

PATENT ABSTRACTS OF JAPAN

(11) Publication number: **07262399 A**(43) Date of publication of application: **13.10.95**

(51) Int. Cl. **G06T 13/00**
G06T 1/00
// G06T 11/40

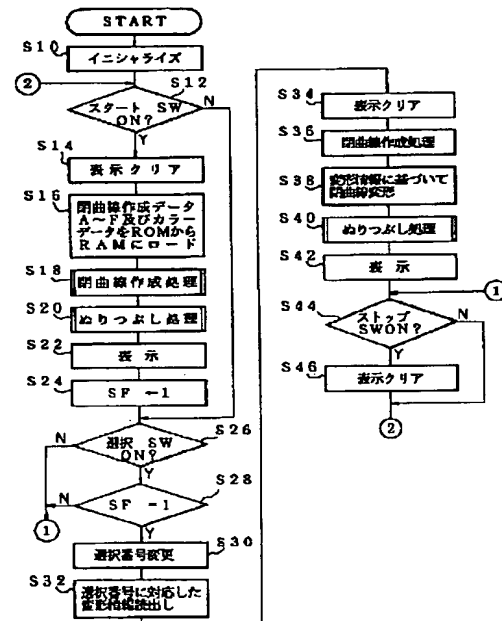
(21) Application number: **06075355**(71) Applicant: **CASIO COMPUT CO LTD**(22) Date of filing: **22.03.94**(72) Inventor: **HAYASHI TETSUYA**(54) **METHOD AND DEVICE FOR PROCESSING IMAGE**

(57) Abstract:

PURPOSE: To change the expression of a portrait or animation while displaying the image of the portrait or animation in colors at low cost with a small amount of data.

CONSTITUTION: When changing the expression of the color portrait, first of all, a basic image is prepared by plural closed curves omitting the colors (step S18) and next, a color basic image is prepared by painting out the boundary and inside of closed curves consisting of the prepared basic image (step S20). Afterwards, deformation information is selected by changing a select number while observing this color basic image (step S30), the closed curves of the basic image are deformed corresponding to the deformation information (step S38) and next, a color deformed image is prepared by painting out the deformed closed curves in the colors. Thus, the expression of the color basic image is changed. Then, the expression of the portrait can be changed only by simply providing closed curve data, color data and deformation data for changing the expression of the portrait.

COPYRIGHT: (C)1995,JPO



(11)特許出願公開番号

特開平7-262399

(43)公開日 平成7年(1995)10月13日

技術表示箇所

1/00

340 D

3 1 0 A

審査請求 未請求 請求項の数14 FD (全 24 頁) 最終頁に続く

(71)出願人 000001443

カシオ計算機株式会社

東京都新宿区西新宿2丁目6番1号

(72) 発明者 林 哲也

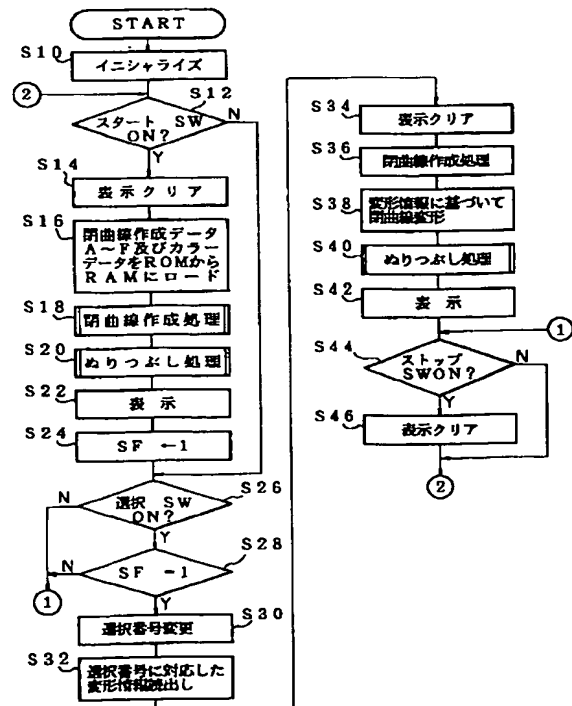
東京都羽村市栄町3丁目2番1号 カシオ
計算機株式会社羽村技術センター内

(54) 【発明の名称】 画像処理方法およびその装置

(57) 【要約】

【目的】 少ないデータで、低コストに、似顔絵やアニメーション等の画像をカラー表示させつつ、その表情を変化させる。

【構成】 カラーの似顔絵でその表情を変化させる場合、まず色を省略した複数の閉曲線で基本画像を作成し（ステップS18）、次いで、作成した基本画像を構成する閉曲線の境界および内部を塗りつぶしてカラーの基本画像を作成し（ステップS20）、次いで、このカラーの基本画像を見ながら選択番号を変更して変形情報を選択し（ステップS30）、基本画像の閉曲線を変形情報に応じて変形し（ステップS38）、次いで、変形した閉曲線を色で塗りつぶしてカラーの変形画像を作成し、これによりカラーの基本画像の表情を変化させる。そして、単に閉曲線データ、カラーデータおよび似顔絵の表情を変化させる変形データ（を持つだけで、似顔絵の表情を変化させることを可能にする。



【特許請求の範囲】

【請求項 1】 所定の基本画像をその色が省略され、かつ所定の優先順位を有する複数の閉曲線を組み合わせて作成し、

作成した基本画像を変形データに応じて変形して変形画像を作成し、

この変形画像を構成する各閉曲線の境界および内部を優先順位に従って所定の色で塗りつぶしてカラーの変形画像を作成することを特徴とする画像処理方法。

【請求項 2】 所定の基本画像をその色が省略され、かつ所定の優先順位を有する複数の閉曲線を組み合わせて作成し、

作成した基本画像に対して、その少なくとも一部を修正枠で指定し、

修正枠で指定した基本画像の一部を、その修正枠に対応した変形データに応じて変形し、

変形した基本画像における閉曲線の境界および内部を優先順位に従って所定の色で塗りつぶして、カラーの変形画像を作成することを特徴とする画像処理方法。

【請求項 3】 前記閉曲線は、直線あるいはベジェ曲線のうちの 1 つ以上によって作成されることを特徴とする請求項 1 又は 2 記載の画像処理方法。

【請求項 4】 前記所定の画像を色を省略した複数の閉曲線で作成する処理では、複数の閉曲線のそれぞれを作成するために各閉曲線の所定の座標データからなる閉曲線データを用い、

これら複数の閉曲線の境界およびその内部を指定された色で塗りつぶす処理では、塗りつぶしの色を指定するためのカラーデータを各閉曲線毎に有することを特徴とする請求項 1 又は 2 記載の画像処理方法。

【請求項 5】 前記閉曲線を直線によって作成する処理は、直線を多数のドットによって形成するものであり、直線の両端を形成する 2 つのドット位置を決定するとともに、そのドット位置の傾きを算出する処理と、

この傾きに応じて x 座標又は y 座標を 1 単位ずつ増加させた場合の誤差を累積する処理と、

この誤差が所定値を超えるとときにのみ、次のドットの x 座標又は y 座標の位置を変更するとともに、前記累算された誤差から一定値を減算する処理と、を有することを特徴とする請求項 1 又は 2 記載の画像処理方法。

【請求項 6】 前記基本画像を変形するための変形データは、予め複数設けられており、少なくともその中の 1 つの変形データを選択して基本画像を変形することを特徴とする請求項 1 又は 2 記載の画像処理方法。

【請求項 7】 所定の基本画像をその色が省略され、かつ所定の優先順位を有する複数の閉曲線を組み合わせて作成する閉曲線作成手段と、

この閉曲線作成手段にて作成された基本画像を構成する少なくとも一部の閉曲線を変形させるための変形データ

を設定する変形データ設定手段と、

上記閉曲線作成手段により作成された基本画像をこの変形データ設定手段にて設定された変形データに応じて変形して変形画像を作成する変形手段と、

この変形手段にて作成された変形画像を構成する各閉曲線の境界および内部を優先順位に従って所定の色で塗りつぶして、カラーの変形画像を作成する画像作成手段と、を備えたことを特徴とする画像処理装置。

【請求項 8】 所定の基本画像をその色が省略され、かつ所定の優先順位を有する複数の閉曲線を組み合わせて作成する閉曲線作成手段と、

基本画像の閉曲線を変形させる変形データを設定する変形データ設定手段と、

上記閉曲線作成手段にて作成した基本画像に対して、その少なくとも一部を上記変形データ設定手段にて設定された変形データに対応する修正枠で指定する修正枠指定手段と、

上記修正枠で指定された基本画像の少なくとも一部を上記変形データ設定手段で設定された変形データに応じて変形する画像変形手段と、

この画像変形手段にて変形した基本画像における各閉曲線の境界および内部を優先順位に従って所定の色で塗りつぶして、カラーの変形画像を作成する画像作成手段と、を備えたことを特徴とする画像処理装置。

【請求項 9】 前記閉曲線作成手段は、直線あるいはベジェ曲線のうちの 1 つ以上によって閉曲線を作成することを特徴とする請求項 7 又は 8 記載の画像処理装置。

【請求項 10】 前記閉曲線作成手段は、所定の基本画像を構成する複数の閉曲線のそれぞれを作成するための各閉曲線の所定の座標データからなる閉曲線データによって作成し、

前記カラー画像作成手段は、各閉曲線毎に対応するカラーデータを記憶し、

この記憶されたカラーデータに対応する色で各閉曲線の境界およびその内部を塗りつぶすことを特徴とする請求項 7 又は 8 記載の画像処理装置。

【請求項 11】 前記基本画像、カラー画像あるいは変形画像のうち少なくとも 1 つ以上を表示する表示手段を、さらに有することを特徴とする請求項 7 又は 8 記載の画像処理装置。

【請求項 12】 前記基本画像、カラー画像あるいは変形画像のうち少なくとも 1 つ以上を印刷する印刷手段を、さらに有することを特徴とする請求項 7 又は 8 記載の画像処理装置。

【請求項 13】 前記閉曲線作成手段は、閉曲線を多数のドットからなる直線によって作成する場合、直線の両端を形成する 2 つのドット位置を決定するとともに、そのドット位置の傾きを算出する手段と、この傾きに応じて x 座標又は y 座標を 1 単位ずつ増加させた場合の誤差を累積する手段と、

3

この誤差が所定値を超えるときにのみ、次のドットのx座標又はy座標の位置を変更するとともに、前記累算された誤差から一定値を減算する手段とを有することを特徴とする請求項7又は8記載の画像処理装置。

【請求項14】 前記変形データ設定手段は、基本画像を変形するための変形データを複数記憶している変形データ記憶手段と、

複数の変形データの中から、少なくとも1つの変形データを選択する変形データ選択手段と、を備えていることを特徴とする請求項7又は8記載の画像処理装置。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、画像処理方法およびその装置に係わり、詳しくはアニメーション、似顔絵等の画像の全部又は一部を変形させることによって、画像の表情を変化可能にした画像処理方法およびその方法を実現する装置に関する。

【0002】

【従来の技術】従来、アニメーション、ゲーム等ではビット配列形式の画像データを用いることが多く、この画像データによりキャラクターや背景データを表示している。そして、このビット配列形式の画像データはドットで構成され、かつ各ドット毎に表示色番号あるいはパレット番号を持つようになっている。すなわち、コンピュータ画面上などにおいて、似顔絵やアニメーション画像をカラー表示させるために、ピクセル毎にカラーデータを持っている。

【0003】

【発明が解決しようとする課題】ところで、従来の画像処理装置の場合、コンピュータ画面上あるいはテレビ画面上において、似顔絵やアニメーション等の画像をカラー表示し、かつその表情を変化させるためには、まずカラー表示のためにピクセル毎にカラーデータを持たなければならない、この点から大量のデータが必要であるという問題点があった。また、画像の表情を変化させるためには、例えば似顔絵やアニメーション画像の全部又は一部のデータを、表情の変化数分だけ持つ必要があり、この点からも大量のデータが必要であった。さらに、大量のデータを処理するためのメモリ装置の容量が多くなって、この点でコスト高にもなっていた。

【0004】そこで本発明は、少ないデータで、低コストに、似顔絵やアニメーション画像等の画像をカラー表示させつつ、その表情を変化させることができる画像処理方法およびその装置を提供することを目的としている。

【0005】

【課題を解決するための手段】上記目的達成のため、請求項1記載の発明による画像処理方法は、所定の基本画像をその色が省略され、かつ所定の優先順位を有する複数の閉曲線を組み合わせて作成し、作成した基本画像を

4

形データに応じて変形して変形画像を作成し、この変形画像を構成する各閉曲線の境界および内部を優先順位に従って所定の色で塗りつぶしてカラーの変形画像を作成することを特徴とする。請求項2記載の発明による画像処理方法は、所定の基本画像をその色が省略され、かつ所定の優先順位を有する複数の閉曲線を組み合わせて作成し、作成した基本画像に対して、その少なくとも一部を修正枠で指定し、修正枠で指定した基本画像の一部を、その修正枠に対応した変形データに応じて変形し、変形した基本画像における閉曲線の境界および内部を優先順位に従って所定の色で塗りつぶして、カラーの変形画像を作成することを特徴とする。

【0006】また、好ましい態様として、例えば請求項3記載のように、前記閉曲線は、直線あるいはベジェ曲線のうちの1つ以上によって作成されるようにしてもよい。例えば請求項4記載のように、前記所定の画像を色を省略した複数の閉曲線で作成する処理では、複数の閉曲線のそれぞれを作成するために各閉曲線の所定の座標データからなる閉曲線データを用い、これら複数の閉曲線の境界およびその内部を指定された色で塗りつぶす処理では、塗りつぶすの色を指定するためのカラーデータを各閉曲線毎に有するようにしてもよい。例えば請求項5記載のように、前記閉曲線を直線によって作成する処理は、直線を多数のドットによって形成するものであり、直線の両端を形成する2つのドット位置を決定するとともに、そのドット位置の傾きを算出する処理と、この傾きに応じてx座標又はy座標を1単位ずつ増加させた場合の誤差を累積する処理と、この誤差が所定値を超えるときにのみ、次のドットのx座標又はy座標の位置を変更するとともに、前記累算された誤差から一定値を減算する処理と、を有するようにしてもよい。

【0007】例えば請求項6記載のように、前記基本画像を変形するための変形データは、予め複数設けられており、少なくともその中の1つの変形データを選択して基本画像を変形するようにしてもよい。

【0008】請求項7記載の画像処理装置は、所定の基本画像をその色が省略され、かつ所定の優先順位を有する複数の閉曲線を組み合わせて作成する閉曲線作成手段と、この閉曲線作成手段にて作成された基本画像を構成する少なくとも一部の閉曲線を変形させるための変形データを設定する変形データ設定手段と、上記閉曲線作成手段により作成された基本画像をこの変形データ設定手段にて設定された変形データに応じて変形して変形画像を作成する変形手段と、この変形手段にて作成された変形画像を構成する各閉曲線の境界および内部を優先順位に従って所定の色で塗りつぶして、カラーの変形画像を作成する画像作成手段と、を備えたことを特徴とする。請求項8記載の画像処理装置は、所定の基本画像をその色が省略され、かつ所定の優先順位を有する複数の閉曲線を組み合わせて作成する閉曲線作成手段と、基本画像の

閉曲線を変形させる変形データを設定する変形データ設定手段と、上記閉曲線作成手段にて作成した基本画像に対して、その少なくとも一部を上記変形データ設定手段にて設定された変形データに対応する修正枠で指定する修正枠指定手段と、上記修正枠で指定された基本画像の少なくとも一部を上記変形データ設定手段で設定された変形データに応じて変形する画像変形手段と、この画像変形手段にて変形した基本画像における各閉曲線の境界および内部を優先順位に従って所定の色で塗りつぶして、カラーの変形画像を作成する画像作成手段と、を備えたことを特徴とする。

【0009】また、例えば請求項9記載のように、前記閉曲線作成手段は、直線あるいはベジェ曲線のうちの1つ以上によって閉曲線を作成するようにしてもよい。例えば請求項10記載のように、前記閉曲線作成手段は、所定の基本画像を構成する複数の閉曲線のそれぞれを作成するための各閉曲線の所定の座標データからなる閉曲線データによって作成し、前記カラー画像作成手段は、各閉曲線毎に対応するカラーデータを記憶し、この記憶されたカラーデータに対応する色で各閉曲線の境界およびその内部を塗りつぶすようにしてもよい。例えば請求項11記載のように、前記基本画像、カラー画像あるいは変形画像のうち少なくとも1つ以上を表示する表示手段を、さらに有するようにしてもよい。例えば請求項12記載のように、前記基本画像、カラー画像あるいは変形画像のうち少なくとも1つ以上を印刷する印刷手段を、さらに有するようにしてもよい。

【0010】例えば請求項13記載のように、前記閉曲線作成手段は、閉曲線を多数のドットからなる直線によって作成する場合、直線の両端を形成する2つのドット位置を決定するとともに、そのドット位置の傾きを算出する手段と、この傾きに応じてx座標又はy座標を1単位ずつ増加させた場合の誤差を累積する手段と、この誤差が所定値を超えるとときにのみ、次のドットのx座標又はy座標の位置を変更するとともに、前記累算された誤差から一定値を減算する手段とを有するようにしてもよい。例えば請求項14記載のように、前記変形データ設定手段は、基本画像を変形するための変形データを複数記憶している変形データ記憶手段と、複数の変形データの中から、少なくとも1つの変形データを選択する変形データ選択手段と、を備えるようにしてもよい。

【0011】

【作用】本発明では、カラーの画像（例えば、似顔絵あるいはアニメーション画像）を作成してその表情を変化させる場合、まず色を省略した複数の閉曲線で基本画像を作成し、次いで、作成した基本画像を変形データに応じて変形して変形画像を作成し、この変形画像を構成する各閉曲線の境界および内部を優先順位に従って所定の色で塗りつぶしてカラーの変形画像を作成し、これによりカラーの基本画像の表情を変化させる。

【0012】したがって、似顔絵の画像をカラー表示させてその表情を変化させる場合でも、従来のようにピクセル毎にカラーデータを持つ必要がなく、単に閉曲線データとカラーデータとを持てば画像をカラーで作成することが可能になるとともに、さらに従来のように似顔絵等の全部又は一部のデータを表情の変化数分だけ持つ必要がなく、単に閉曲線データ、カラーデータおよび似顔絵の表情を変化させる変形データ（表情の変化に対応する変形データ）を持てば、似顔絵等の表情を変化させることが可能になる。その結果、データ量が少なく、メモリ装置の容量を少なくしつつ、かつ低コストで似顔絵やアニメーション等の画像をカラーで作成して表示し、かつその表情を変化させることができる。また、他の請求項記載の発明では、カラーの画像を作成してその表情を変化させる場合、まず色を省略した複数の閉曲線で基本画像を作成し、次いで、作成した基本画像に対して、その一部を修正枠で指定し、次いで、修正枠で指定した基本画像の一部を変形データに応じて変形し、さらに変形した基本画像を構成する閉曲線の境界および内部を塗りつぶして、カラーの変形画像を作成することにより、カラーの基本画像の表情を変化させる。したがって、画像の一部を修正枠で指定できるから、修正枠で指定した基本画像の一部の表情を変化させることができる。

【0013】

【実施例】以下、図面を参照して本発明の実施例について説明する。

似顔絵作成装置の構成

図1は本発明に係る画像処理方法を実現する似顔絵作成装置（画像処理装置に相当）の第1実施例を示す構成図である。図1において、似顔絵作成装置は大きく分けてCPU1、ROM2、RAM3、スイッチ部4、表示部5、カラープリンタ6および各部を結ぶバス7によって構成されている。CPU1は装置全体を制御するもので、スイッチ部4から描画指令が入力されると、その指令情報に対応すべくROM2に格納されている制御プログラムに基づいて似顔絵を作成するために必要な処理（例えば、似顔絵の輪郭や各部位の描画に必要な直線又はベジェ曲線のデータ計算、塗りつぶしの処理、図形変形処理およびデータの補間処理）を行う。また、CPU1は内部レジスタを有しており、内部レジスタにフラグやポインタの値を格納する。

【0014】ROM2はCPU1の制御プログラムを格納しているとともに、後述の図3に示すように似顔絵の輪郭や顔の各部位を表す閉曲線作成データ、閉曲線カラーデータ、変形データ、修正枠データ等のパラメータを予め格納している。RAM3はCPU1が演算処理を行う際のワークエリアとして用いられ、データを一時的に記憶する。スイッチ部4はオペレータによって操作されるもので、似顔絵の表情を選択するときに操作される選択スイッチ（変形データ選択手段）、似顔絵作成処理を

開始するときに操作されるスタートスイッチおよび似顔絵の表示を停止させるときに操作されるストップスイッチを有している。なお、各操作スイッチは単独操作のプッシュスイッチでもよいし、あるいは複数のスイッチからなるスイッチボード、キーボード等でもよい。また、スイッチ部4としてスイッチボード等の他に、マウス、トラックボール等を用いてもよい。

【0015】表示部（表示手段）5はCPU1の演算処理によって作成される似顔絵をそのプロセス毎に表示するもので、例えば画像信号発生回路（Video Display Processer：以下VDPという）、VRAM、TVディスプレイを含んで構成される。なお、画面の表示はTVディスプレイではなく、カラー表示が可能なものであれば、LCDでもよい。カラープリンタ（印刷手段）6は所定の用紙に画像を印刷するもので、例えば表示部5に表示された似顔絵をカラーで印刷することが可能である。上記CPU1、ROM2およびRAM3は全体として閉曲線作成手段21、変形手段22、画像作成手段23を構成する。ROM2は変形データ記憶手段を構成する。ROM2およびスイッチ部4の選択スイッチは、全体として変形データ設定手段としての機能を有する。

【0016】次に、作用を説明する。

メインプログラム

図2は似顔絵作成処理のメインプログラムを示すフローチャートである。このプログラムがスタートすると、まずステップS10でイニシャライズ（初期化处理）を行う。これにより、例えばCPU1内の各種レジスタのイニシャライズ、RAM3のワークエリアのクリア、サブルーチンのイニシャライズ、フラグのリセット等が行われる。次いで、ステップS12でスイッチ部4のスタートスイッチがオンしているか否かを判別する。スタートスイッチがオンしていれば、ステップS14で表示部5の表示をクリアする。ここで最初のルーチンでは、例えば初期画面を表示させてもよいし、あるいは何等かの操作画面を表示してもよいし、あるいは何も表示させなくてもよいが、何れにしても、ステップS14で一旦表示がクリアされる。また、例えば既に似顔絵が表示されている場合には、以前の似顔絵がクリアされる。

【0017】ここで、ROM2には図3に示すように基準となる似顔絵（以下、適宜基本画像という）を作成するための似顔絵データおよびその似顔絵の表情を変化させるための複数の変形データ（1）～（n）が格納されている。似顔絵データは似顔絵の輪郭や顔の各部位を表す閉曲線作成データA～Fと、閉曲線の色を指定する閉曲線カラーデータA～Fとに区分されて所定のエリアに格納されている。なお、閉曲線カラーデータA～Fは、閉曲線作成データA～Fのそれぞれの色（閉曲線の境界および内部を塗りつぶす色）を指定するものである。また、複数の変形データ（1）～（n）は基準となる似顔絵の表情をどのように変化させるかを示すデータであ

り、それぞれ予め設定されて所定のエリアに格納されている。例えば、変形データ（1）は怒った表情、変形データ（2）は笑った表情という具合に、似顔絵の表情を変化させるものであり、選択スイッチによって選択が可能である。なお、この場合に似顔絵をどのような表情で表示させるかにより、変形データを自由に設定することが可能であるとともに、変形データの内容（変形情報）も自由に設定可能である。

【0018】再びプログラムに戻り、次いで、ステップS16で基本となる似顔絵を作成するために、閉曲線作成データA～Fおよび閉曲線カラーデータA～FをROM2から読み出して、RAM3にロードする。ここで、RAM3には図4に示すような各種のワークエリアが設けられており、ROM2から読み出されたデータは対応するエリアにロードされる。RAM3のワークエリアを説明すると、以下のようになっている。

閉曲線A作成データ……閉曲線Aを作成するためのデータを格納するエリア

閉曲線B作成データ……閉曲線Bを作成するためのデータを格納するエリア

閉曲線C作成データ……閉曲線Cを作成するためのデータを格納するエリア

閉曲線D作成データ……閉曲線Dを作成するためのデータを格納するエリア

閉曲線E作成データ……閉曲線Eを作成するためのデータを格納するエリア

閉曲線F作成データ……閉曲線Fを作成するためのデータを格納するエリア

【0019】閉曲線Aカラーデータ……閉曲線Aを塗りつぶす色（カラー）を指定するデータを格納するエリア
閉曲線Bカラーデータ……閉曲線Bを塗りつぶす色（カラー）を指定するデータを格納するエリア

閉曲線Cカラーデータ……閉曲線Cを塗りつぶす色（カラー）を指定するデータを格納するエリア

閉曲線Dカラーデータ……閉曲線Dを塗りつぶす色（カラー）を指定するデータを格納するエリア

閉曲線Eカラーデータ……閉曲線Eを塗りつぶす色（カラー）を指定するデータを格納するエリア

閉曲線Fカラーデータ……閉曲線Fを塗りつぶす色（カラー）を指定するデータを格納するエリア

【0020】カラー状態フラグ（A）……閉曲線Aをカラー設定エリアに指定（つまり、色で塗りつぶす）するか否かを決定するフラグを格納するエリア

カラー状態フラグ（B）……閉曲線Bをカラー設定エリアに指定（つまり、色で塗りつぶす）するか否かを決定するフラグを格納するエリア

カラー状態フラグ（C）……閉曲線Cをカラー設定エリアに指定（つまり、色で塗りつぶす）するか否かを決定するフラグを格納するエリア

カラー状態フラグ（D）……閉曲線Dをカラー設定エリ

アに指定（つまり、色で塗りつぶす）するか否かを決定するフラグを格納するエリア

カラー状態フラグ（E）……閉曲線Eをカラー設定エリアに指定（つまり、色で塗りつぶす）するか否かを決定するフラグを格納するエリア

カラー状態フラグ（F）……閉曲線Fをカラー設定エリアに指定（つまり、色で塗りつぶす）するか否かを決定するフラグを格納するエリア

【0021】バックグラウンドカラー番号……背景となる部分を塗りつぶす色（カラー）を指定するデータを格納するエリア

選択番号……選択スイッチによって選択された変形データの番号を格納するエリア

変形データ……選択スイッチによって選択された似顔絵の表情を変化させる変形データを格納するエリア

また、その他に作成した似顔絵の各部位に対応する閉曲線を格納するエリア11～16がある。例えば、エリア11は前髪を格納するエリア、エリア12は髪型を格納するエリア、エリア13は髪における光沢部分を格納するエリア、エリア14は顔の輪郭を格納するエリア、エリア15は顔のパーツを格納するエリア、エリア16は首を格納するエリアである。

【0022】再びフローチャートの説明に戻り、ステップS16を経ると、続くステップS18に進み、ロードした閉曲線作成データA～Fに基づいて閉曲線A～Fをそれぞれ作成する処理（詳細はサブルーチンで後述）を行うとともに、ステップS20では作成した閉曲線A～Fを塗りつぶす処理（詳細はサブルーチンで後述）を行う。これにより、基本画像の各部位に対応する閉曲線A～Fが作成されるとともに、作成された各閉曲線A～Fを所定の色で塗りつぶす処理が行われ、最終的にカラーの基本画像が作成される。次いで、ステップS22で表示処理を行う。これにより、作成されたカラーの基本画像（似顔絵）が表示部5に表示される。次いで、ステップS24でスタートフラグSFを「1」にセットする。スタートフラグSFは、イニシャライズ処理で「0」にクリアされるとともに、スタートスイッチのオン操作により基本画像が作成されたときに、「1」にセットされるものである。そして、SF＝「1」になると、基本画像の表情を変化させることが可能になる。

【0023】ステップS24でSF＝「1」になると、ステップS26に進む。一方、上記ステップS12でスタートスイッチがオンしていなければ、ステップS12でNOに分岐しステップS26に直接ジャンプする。したがって、このときはカラーの基本画像が作成されない。ステップS26では選択スイッチがオンしているか否かを判別する。選択スイッチは基本画像の表情を変化させる場合に、その変形データを選択するときに操作されるものであるから、選択スイッチがオンしていれば後述のステップで選択番号を変更することになる。また、

選択スイッチがオンしていなければ、NOに分岐してステップS44にジャンプする。

【0024】A. 選択スイッチがオンしていないとき
ステップS26からNOに分岐してステップS44にジャンプし、ステップS44でストップスイッチがオンしているか否かを判別する。ストップスイッチは似顔絵の表示を停止させるときに操作されるものであるから、ストップスイッチがオンしていると、ステップS46に進んで画像の表示をクリアする。これにより、表示部5における似顔絵画像の表示が停止する。次いで、ステップS12に戻ってスタートスイッチの操作を判別する。また、スイッチ44でストップスイッチがオンしていなければ、ステップS46をジャンプしてステップS12に戻り、同様にスタートスイッチの操作を判別する。したがって、このときは表示部5における似顔絵画像の表示は停止せず、そのまま画像の表示が継続する。例えば、同じ表情の似顔絵を表示しておくような場合である。

【0025】B. 選択スイッチがオンしたとき
ステップS26からYESに分岐し、続くステップS28でスタートフラグSFが「1」であるか否かを判別する。スタートフラグSFが「0」のときはステップS44にジャンプする。したがって、このときは選択スイッチをオン操作しても似顔絵の変形は行われない。すなわち、基本画像が作成されていなければ、画像の変形ができないからである。ステップS28でスタートフラグSFが「1」のときはステップS30に進んで変形データを選択する選択番号を変更する。例えば、前回のルーチンで変形データ（1）が選択されていた場合には、選択スイッチの操作により次の変形データ（2）に移り、選択番号＝2とする。

【0026】次いで、ステップS32で今回のルーチンの選択番号に対応した変形情報をROM2から読み出し、RAM3にロードする。例えば、選択番号＝2であれば、変形データ（2）という情報が読み出される。次いで、ステップS34で一旦表示部5の表示画像をクリアする。これにより、いままで表示されていた似顔絵が消える。次いで、ステップS36で閉曲線作成データA～Fに基づいて閉曲線A～Fをそれぞれ作成する処理（詳細はサブルーチンで後述）を行うとともに、ステップS38では今回読み出した変形情報に基づいて閉曲線を変形する処理を行う。これにより、基本画像の一部が読み出された変形情報（例えば、変形データ（2）という情報）に従って変形し、基本画像の表情が変化することになる。次いで、ステップS40で閉曲線によって作成された変形した表情の基本画像（色はまだ無い）を塗りつぶす処理（詳細はサブルーチンで後述）を行う。これにより、表情の変化した基本画像の各閉曲線に対して色を付ける処理が行われ、最終的にカラーの似顔絵画像が作成される。次いで、ステップS42で表示処理を行う。したがって、基本画像を変形してその表情が変化

したカラーの似顔絵が表示部5に表示される。

【0027】次いで、ステップS44に進み、前述したようにストップスイッチがオンしているか否かを判別する。ストップスイッチがオンしていなければ、ステップS46をジャンプしてステップS12に戻る。したがって、このときは表示部5における似顔絵画像の表示は停止せず、そのまま画像の表示が継続する。例えば、同じ表情の似顔絵を表示しておく場合である。一方、似顔絵を消す場合には、ストップスイッチを押せば、このステップS44からYESに分岐してステップS46で表示をクリアする。ステップS46を経ると、ステップS12に戻る。このように、カラーの基本画像が一旦作成して表示され、その画像を見ながら選択スイッチを操作することにより、基本画像を変形してその表情を変化させることが行われる。この場合、変形データをいろいろ選択することにより、例えば笑った似顔絵、怒った似顔絵等の様々な表情に変化させることができる。

【0028】閉曲線A～Fの作成処理のサブルーチン

図5はメインプログラムの閉曲線A～Fの作成処理（ステップS18、ステップS36）のサブルーチンを示すフローチャートである。このサブルーチンに移行すると、ステップS50で閉曲線Aを作成する処理を行う。例えば、閉曲線Aが前髪に相当する場合には、前髪の閉曲線（図4のエリア11に格納される画像）が作成される。次いで、ステップS52で閉曲線Bを作成する処理を行う。例えば、閉曲線Bが髪型に相当する場合には、髪型の閉曲線（図4のエリア12に格納される画像）が作成される。次いで、ステップS54で閉曲線Cを作成する処理を行う。例えば、閉曲線Cが髪における光沢部分に相当する場合には、光沢部分の閉曲線（図4のエリア13に格納される画像）が作成される。次いで、ステップS56で閉曲線Dを作成する処理を行う。例えば、閉曲線Dが顔の輪郭に相当する場合には、輪郭の閉曲線（図4のエリア14に格納される画像）が作成される。

【0029】次いで、ステップS58で閉曲線Eを作成する処理を行う。例えば、閉曲線Eが顔のパーツに相当する場合には、顔のパーツの閉曲線（図4のエリア15に格納される画像）が作成される。なお、顔のパーツとしては、眉毛、目、鼻、口がある。次いで、ステップS60で閉曲線Fを作成する処理を行う。例えば、閉曲線Fが首に相当する場合には、首の閉曲線（図4のエリア16に格納される画像）が作成される。ステップS60を経ると、メインプログラム（あるいはタイミントラプト処理）にリターンする。このようにして、似顔絵を作成するための各閉曲線A～Fが作成される。

【0030】ドット算出処理のサブルーチン

図6、図7は閉曲線A～Fを作成する場合のドット算出処理のサブルーチンを示すフローチャートである。この処理は、2点間を結ぶ直線を表示部5のTVディスプレイ（例えば、コンピュータ画面やテレビ画面に相当）に

表示させるために行うもので、2点間を結ぶ直線は多数のドットを算出して結ぶことによって表示される。なお、この技術はコンピュータの直線描画に用いられているいわゆる「整数型Bresenham」のアルゴリズムを応用したものである。通常の「Bresenham」のアルゴリズムでは、直線の傾きや誤差用の判定に、浮動小数点の加減算と除算を行う必要があるが、整数計算を用いるとともに、さらに除算を無くしてアルゴリズムの処理速度を高めたのが、「整数型Bresenham」のアルゴリズムである。なお、以下のフローチャートの説明では、コンピュータで使用される「C言語」の記述方式を用いており、各ステップの内容は必要に応じて「C言語」の記述で表すこととする。これは、後述の各フローチャートについても、同様である。

【0031】このサブルーチンでは、まずステップS100で直線の始点、終点の座標の差（すなわち、傾き）を演算する。直線の始点、終点はx座標およびy座標を用いて表される。例えば、直線の始点、終点をそれぞれ（ x_1 、 y_1 ）および（ x_2 、 y_2 ）とし、互いに等しくないものとする。また、全ての変数は整数とする。そして、

$$x = x_1$$

$$y = y_1$$

$$\Delta x = x_2 - x_1$$

$\Delta y = y_2 - y_1$ とし、アルゴリズムの誤差項eに対して $e = 2e\Delta x$ （初期値 $= -1/2$ ）として演算を進め、直線の傾きが $1/2$ 以上の場合は1単位だけ離れた次のポイントに変数yを増加させ、直線の傾きが $1/2$ 未満の場合は次のポイントに変数yを増加させないようにする。なお、1単位だけ離れた次のポイントに変数yを増加させた場合には、次のピクセルを決定する前に誤差項eを再度初期化するために、誤差項eから「1」を減算する。

【0032】さて、ステップS100では直線の始点、終点の座標の差として、x座標については、gainxを算出し、y座標については、gainyを算出する。この場合、開始点のx座標はfromx、開始点のy座標はfromy、終点のx座標はto.x、終点のy座標はto.yで表している。そして、x座標についての差の演算式は、終点のx座標to.xから開始点のx座標fromxを減算したもの、つまり（to.x-fromx）として表される。同様に、y座標についての差の演算式は、終点のy座標to.yから開始点のy座標fromyを減算したもの、つまり（to.y-fromy）として表される。次いで、ステップS102で直線の始点、終点の座標の差（すなわち、傾き）の絶対値 Δ （delta）をそれぞれx座標およびy座標について演算する。例えば、x座標では $\text{delta}x = \text{abs}(\text{gain}x)$ として演算する。delta xは $\Delta x = x_2 - x_1$ に相当する。y座標では $\text{delta}y = \text{abs}(\text{gain}y)$ として演算する。delta yは $\Delta y = y_2 - y_1$ に相当する。

【0033】次いで、ステップS104で直線の始点、終点の座標のx座標の差gainxの符号を判別する。gainxが正のときはステップS106に進んでratex=1にセットし、gainxが「0」のときはステップS108に進んでratex=0にセットし、さらにgainxが負のときはステップS110に進んでratex=-1にセットする。ステップS106～ステップS110の何れかを経ると、ステップS112に進む。ステップS112では同様に直線の始点、終点の座標のy座標の差gainyの符号を判別する。gainyが正のときはステップS114に進んでratey=1にセットし、gainyが「0」のときはステップS116に進んでratey=0にセットし、さらにgainyが負のときはステップS118に進んでratey=-1にセットする。ステップS114～ステップS118の何れかを経ると、ステップS120に進む。

【0034】ステップS120では直線のx座標方向の傾きの絶対値deltaxと、y座標方向の傾きの絶対値deltayとを比較し、deltaxがdeltayより大きいかなんかを判別する。これは、x座標方向とy座標方向のどちらをメインの誤差項にするかを決定するためである。deltax>deltayのとき（すなわち、TRUEのとき、いわゆるYESに相当）は、ステップS122に進んでメイン誤差項（main delta）をdeltaxにする。なお、図面上では（main_delta）としてアンダーバーを付加しているが、本文中では繁雑になるので、このアンダーバーを省略する。

【0035】次いで、ステップS124でサブ誤差項（sub delta）をdeltayにする。なお、図面上では（sub_delta）としてアンダーバーを付加しているが、本文中では繁雑になるので、このアンダーバーを省略する。次いで、ステップS126でメイン決定フラグ（flag）をTRUEにして図7のステップS134に進む。メイン決定フラグ（flag）はメイン誤差項（main delta）をdeltaxにしたか、あるいはdeltayにしたかを示すものである。一方、deltax≤deltayのとき（すなわち、FALSEのとき、いわゆるNOに相当）は、ステップS128に進んでメイン誤差項（main delta）をdeltayにする。次いで、ステップS130でサブ誤差項（sub delta）をdeltaxにする。次いで、ステップS132でメイン決定フラグ（flag）をFALSEにして図7のステップS134に進む。

【0036】図7に移り、ステップS134では直線の始点のx座標に相当するドットdotx[0]を開始点のx座標fromxにセットするとともに、直線の始点のy座標に相当するドットdoty[0]を開始点のy座標fromyにセットする。これにより、始点のドットの各座標が決定される。次いで、ステップS136で同じくx座標方向のパラメータpxを開始点のx座標fromxにセットするとともに、y座標方向のパラメータpyを開始点のy座標fromyにセットする。次いで、ステップS138で誤差項

eを $error = 2 \times (sub\ delta) - (main\ delta)$ なる式から求める。この式によると、例えば $e = 2 \times \Delta y - \Delta x$ のようになる。

【0037】次いで、ステップS140でポインタiを初期値[0]にセットする。ポインタiは直線を形成するドットを順次指定していくもので、[1]なる整数によってインクリメントされる。次いで、ステップS142でポインタiがメイン誤差項（main delta）より小さいかなんかを判別する。TRUEのときはステップS144に進んで誤差項errorが $error \geq 0$ であるかなんかを判別する。error≥0のときは、ステップS146でメイン決定フラグ（flag）を判別する。メイン決定フラグ（flag）がTRUEであればx座標の方がメイン誤差項（main delta）に設定されているから、ステップS148に進んで今回のルーチンのy座標方向のパラメータpyを次式に従って演算する。

$py = py + ratey$

【0038】次いで、ステップS150で今回のルーチンの誤差項eを次式に従って演算する。

$error = error - 2 \times (main\ delta)$

これにより、誤差項eがメインの誤差によって変化する。次いで、再びステップS144に戻って同様のループを繰り返す。一方、ステップS146でメイン決定フラグ（flag）の判別結果がFALSEであれば、y座標の方がメイン誤差項（main delta）に設定されていると判断して、ステップS144からステップS152に分岐し、今回のルーチンのx座標方向のパラメータpxを次式に従って演算する。

$px = px + ratex$

次いで、ステップS150に進み、今回のルーチンの誤差項eを前記演算式に従って演算する。その後、ステップS144に戻って同様のループを繰り返す。そして、ステップS144で誤差項errorが $error < 0$ になると、ステップS154に抜ける。以上のステップS144～ステップS152の処理により、メイン誤差項（main delta）の処理が行われる。

【0039】次いで、サブ誤差項（sub delta）の処理に移る。まず、ステップS154でメイン決定フラグ（flag）を判別し、メイン決定フラグ（flag）がTRUEであれば、x座標の方がメイン誤差項（main delta）に設定されているから、ステップS156に進んで今回のルーチンのx座標方向のパラメータpxを次式に従って演算する。

$px = px + ratex$

次いで、ステップS160で今回のルーチンの誤差項eを次式に従って演算する。

$error = error + 2 \times (sub\ delta)$

これにより、誤差項eがサブ誤差sub deltaによって変化する。次いで、ステップS162でx座標方向の今回のルーチンのパラメータpxの値を直線のx座標に対応

するドット $\text{dot } x[i+1]$ とする。最初のルーチンでは $i=0$ であるから、 $\text{dot } x[i+1]=\text{dot } x[1]$ となる。つまり、始点から 1 ドットだけ離れた x 座標のドット $\text{dot } x$ が算出される。次いで、ステップ S164 でポインタ i を整数 [1] だけインクリメントし、再びステップ S142 に戻ってループを繰り返す。

【0040】一方、ステップ S154 でメイン決定フラグ (flag) の判別結果が FALSE であれば、 y 座標の方がメイン誤差項 (main delta) に設定されていると判断して、ステップ S154 からステップ S158 に分岐し、今回のルーチンの y 座標方向のパラメータ py を次式に従って演算する。

$$py = py + \text{ratey}$$

その後、ステップ S160 に進み、今回のルーチンのサブ誤差項 e を前記演算式に従って演算し、次いで、ステップ S162、ステップ S164 を経て、再びステップ S142 に戻ってループを繰り返す。そして、順次ポインタ i をインクリメントして、上記ループを繰り返し、ステップ S142 でポインタ i がメイン誤差項 (main delta) 以上になると、このサブルーチンを終了する。このようにして、2 点間を結ぶ直線のドットが算出され、これらのドットを結ぶことによって、2 点間を結ぶ直線を表示させる処理が行われる。この場合、本実施例では「整数型 Bresenham」のアルゴリズムを用いて、整数計算を主に行っているため、アルゴリズムの処理速度が速い。

【0041】ベジェ曲線データ計算処理のサブルーチン 図 8、図 9 はベジェ曲線を作成する場合のデータ計算処理のサブルーチンを示すフローチャートである。ベジェ曲線は 2 点のコントロールポイント (P_2 、 P_3) と、2 点のアンカーポイント (P_1 、 P_4) によって定義され、一般的には次式のように表される。ここで、アンカーポイント (P_1 、 P_4) とは曲線の両端の 2 点で、各点の座標は、例えば $P_1(x_1, y_1)$ 、 $P_4(x_4, y_4)$ のように表される。また、コントロールポイント (P_2 、 P_3) とは曲線の曲がり方を制御する 2 点のことで、各点の座標は、例えば $P_2(x_2, y_2)$ 、 $P_3(x_3, y_3)$ のように表される。

$$B(t) = (1-t)^3 \cdot P_1 + 3t \cdot (1-t)^2 \cdot P_2 + 3t^2 \cdot (1-t) \cdot P_3 + t^3 \cdot P_4$$

ただし、 $0 \leq t \leq 1$

t は 0 から 1 の間の任意に変化するため、0 から 1 の間を細かく分けることによって、ベジェ曲線上の点が精度良く算出される。なお、計算の回数が多くなるため、そこで適当な個数の曲線上の点を算出した後に、それらを直線によって補間する方法を採るようにしている。なお、例えば $t=0$ を上式に代入すると、 $P_1(x_1, y_1)$ となり、これは一方の端点を表す。また、 $t=1$ を上式に代入すると、 $P_4(x_4, y_4)$ となり、これは他方の端点を表す。なお、直線は幅の無い真っ直ぐなベ

ジェ曲線として表示することも可能である。

【0042】まず、ステップ S200 でポインタ i を [0] にセットし、ステップ S202 で変数 t を $t=i/n$ として算出する。ポインタ i は [1] 毎にインクリメントされるものであるため、変数 t を n によって除算することで、0 から 1 の間に細かく分けるものである。次いで、ステップ S204 で偏差 tn を $tn=1-t$ なる式によって演算する。次いで、ステップ S206 でベジェ曲線 $B(t)$ の変数 t の 0 乗の項 (すなわち、 $t=1$) の x 座標について次式に従って演算する。

$$sx[0] = tn \times tn \times tn \times px[0]$$

$$= (1-t)^3 \cdot px[0]$$

$$= (1-t)^3 \cdot P_1(x)$$

$px[0]$ はアンカーポイント P_1 の x 座標である。次いで、ステップ S208 でベジェ曲線 $B(t)$ の変数 t の 1 乗の項 (すなわち、 t) の x 座標について次式に従って演算する。

$$sx[1] = 3 \cdot 0 \times t \times tn \times tn \times px[1]$$

$$= 3t \cdot (1-t)^2 \cdot px[1]$$

$$= 3t \cdot (1-t)^2 \cdot P_2(x)$$

$px[1]$ はコントロールポイント P_2 の x 座標である。

【0043】次いで、ステップ S210 でベジェ曲線 $B(t)$ の変数 t の 2 乗の項 (すなわち、 t^2) の x 座標について次式に従って演算する。

$$sx[2] = 3 \cdot 0 \times t \times t \times tn \times px[2]$$

$$= 3t^2 \cdot (1-t) \cdot px[2]$$

$$= 3t^2 \cdot (1-t) \cdot P_3(x)$$

$px[2]$ はコントロールポイント P_3 の x 座標である。次いで、ステップ S212 でベジェ曲線 $B(t)$ の変数 t の 3 乗の項 (すなわち、 t^3) の x 座標について次式に従って演算する。

$$sx[3] = t \times t \times t \times px[3]$$

$$= t^3 \cdot px[3]$$

$$= t^3 \cdot P_4(x)$$

$px[3]$ はアンカーポイント P_4 の x 座標である。

【0044】次いで、ステップ S214～ステップ S220 において、ベジェ曲線 $B(t)$ の変数 t の y 座標について上記同様の演算を行う。すなわち、ステップ S214 でベジェ曲線 $B(t)$ の変数 t の 0 乗の項 (すなわち、 $t=1$) の y 座標について次式に従って演算する。

$$sy[0] = tn \times tn \times tn \times py[0]$$

$$= (1-t)^3 \cdot py[0]$$

$$= (1-t)^3 \cdot P_1(y)$$

$py[0]$ はアンカーポイント P_1 の y 座標である。次いで、ステップ S216 でベジェ曲線 $B(t)$ の変数 t の 1 乗の項 (すなわち、 t) の y 座標について次式に従って演算する。

$$sy[1] = 3 \cdot 0 \times t \times tn \times tn \times py[1]$$

$$= 3t \cdot (1-t)^2 \cdot py[1]$$

$$= 3t \cdot (1-t)^2 \cdot P_2(y)$$

$py[1]$ はコントロールポイント P_2 の y 座標である。

【0045】次いで、ステップS218でベジェ曲線 $B(t)$ の変数 t の2乗の項(すなわち、 t^2)の y 座標について次式に従って演算する。

$$\begin{aligned} sy[2] &= 3 \cdot 0 \times t \times t \times tn \times py[2] \\ &= 3t^2 \cdot (1-t) \cdot py[2] \\ &= 3t^2 \cdot (1-t) \cdot P_3(y) \end{aligned}$$

$py[2]$ はコントロールポイント P_3 の y 座標である。次いで、ステップS220でベジェ曲線 $B(t)$ の変数 t の3乗の項(すなわち、 t^3)の y 座標について次式に従って演算する。

$$\begin{aligned} sy[3] &= t \times t \times t \times py[3] \\ &= t^3 \cdot py[3] \\ &= t^3 \cdot P_4(y) \end{aligned}$$

$py[3]$ はアンカーポイント P_4 の y 座標である。

【0046】図9に移り、ステップS222でベジェ曲線 $B(t)$ 上のポイントとポイント間を直線によって補間する場合の補間定数の x 座標 $bx[i]$ 、 y 座標 $by[i]$ を最初は共に、 $bx[i] = by[i] = 0$ にセットする。次いで、ステップS224でポインタ j を初期値[0]におき、続くステップS226で補間定数の x 座標 $bx[i]$ および y 座標 $by[i]$ をそれぞれ次式に従って演算する。

$$\begin{aligned} bx[i] &= bx[i] + sx[j] \\ by[i] &= by[i] + sy[j] \end{aligned}$$

次いで、ステップS228でポインタ j が[4]以上になったか否かを判別し、[4]未満のときはステップS230に進んでポインタ j を[1]だけインクリメントし、ステップS226に戻り、ループを繰り返す。そして、ステップS228でポインタ j が[4]以上になると、ステップS232に抜ける。

【0047】次いで、ステップS232でポインタ i が n 以上になったか否かを判別し、 n 未満のときはステップS234に進んでポインタ i を[1]だけインクリメントし、図8のステップS202に戻り、同様のループを繰り返す。そして、ステップS232でポインタ i が n 以上になると、今回のサブルーチンを終了する。このようにして、ポインタ i を[0]から n までインクリメントすることによって2点のコントロールポイント(P_2 、 P_3)と、2点のアンカーポイント(P_1 、 P_4)によって定義されるベジェ曲線 $B(t)$ 上の点を算出し、このとき t を0から1の間に分けて細かく変化させることにより、ベジェ曲線上の点を精度良く算出することができる。また、ベジェ曲線 $B(t)$ 上のポイントとポイント間を4つの点で適切に補間することにより、一層精度が高められる。

【0048】カラー設定処理のサブルーチン

図10、図11はメインプログラムにおける塗りつぶし処理(ステップS20、ステップS40)のうち、カラー設定処理のサブルーチンを示すフローチャートであ

る。この処理は、作成された閉曲線に対して指定された色を塗っていくものである。このサブルーチンでは、まずステップS300でカラーエリアをクリアする。これにより、最初は色を塗る対象となる全てのエリアが一旦クリアされ、無色になる。次いで、ステップS302でカラー状態フラグ $Cflag$ を初期設定する。カラー状態フラグ $Cflag$ とは、対応する閉曲線をカラー設定エリアに指定(つまり、色で塗りつぶす)するか否かを決定するフラグであり、初期設定されると、図12に示すように、すべて[-1]になる。カラー状態フラグ=-1とは、カラーを指定するカラー番号で取り得ない番号に設定することである。

【0049】図12の例では、オブジェクト1のカラー～オブジェクト n (各オブジェクトは、例えば似顔絵の各部位に相当)のカラーが全て初期状態に設定され、バックグラウンドのカラーだけ何等かのカラー番号が入り、色で塗られるようになっている。そして、各オブジェクトのカラー切替フラグが[1]になるたびに、それに対応するカラー状態フラグが反転(例えば、[-1]から[1]になる)するようになっている。ステップS304ではポインタ i を[0]にクリアし、ステップS306でポインタ j を[0]にクリアする。ポインタ i は画面上のライン(例えば、0から524までのラインがある)を順次指定するもので、ポインタ j はライン上のドットを順次指定するものである。ポインタ i =[0]にすることによりライン0が指定され、ポインタ j =[0]にすることによりライン0上のドット0が指定される。

【0050】次いで、ステップS308でラインカラー $lcolor$ を[-1]に設定するとともに、ラインナンバー(ライン番号) $lnum$ を[-1]に設定する。ラインカラー $lcolor$ は閉曲線のカラーを指定するものである。また、ラインナンバー $lnum$ は閉曲線の境界線の番号を指定するものである。何れも、[-1]に設定されることにより、初期状態となる。次いで、ステップS310で閉曲線ナンバー k を($n-1$)に設定する。例えば、閉曲線が6個あるとすると、閉曲線ナンバー k を[5]に設定する。閉曲線ナンバー k は値が小さい程、優先度が高くなるものである。したがって、 $k=0$ が最も優先度が高く、以下、順次低くなる。これにより、ステップS310では2番目に優先度の低い状態に設定されることになる。

【0051】次いで、ステップS312で閉曲線関数 $C[k][i][j]$ を判断する。閉曲線関数 $C[k][i][j]$ は閉曲線ナンバー k の優先度と、今回の閉曲線が何れかの閉曲線の中にあるか(つまり、何れかの閉曲線の境界線が検出されたか)否かを判断可能なものである。閉曲線関数 $C[k][i][j]$ の判断結果がTRUEのときは、ステップS314に進んでカラー状態フラグ $Cflag[k]$ を反転する。これにより、今回の

閉曲線ナンバー k によって指定された閉曲線のカラー状態フラグが $[-1]$ から反転して $[1]$ になる。したがって、対応する閉曲線に対して色が塗られることになる。次いで、ステップ $S 3 1 6$ でラインカラー $lcolor$ にカラー番号を入れるとともに、ラインナンバー $lnum$ に閉曲線ナンバー k を入れる。これにより、今回の閉曲線に対してカラー番号（塗るべき色を指定するもの）が与えられるとともに、その閉曲線の優先度が付与される。次いで、ステップ $S 3 1 8$ で閉曲線ナンバー k を $[1]$ だけデクリメントする。これにより、1つだけ優先度が高くなる。

【0052】一方、ステップ $S 3 1 2$ で閉曲線関数 $C[k][i][j]$ の判断結果が $FALSE$ のときは、ステップ $S 3 1 4$ 、ステップ $S 3 1 6$ をジャンプしてステップ $S 3 1 8$ に進む。したがって、このときは直ちに閉曲線ナンバー k が $[1]$ だけデクリメントされ、1つ上の優先度に移行することになる。ステップ $S 3 1 8$ を経ると、続くステップ $S 3 2 0$ で閉曲線ナンバー k が $[0]$ 未満である（つまり、未だ一番高い優先度にはなっていない状態である）か否かを判別する。 $k=0$ でなければ（最初のルーチンでは優先度は最高になっていない）、 $TRUE$ に分岐してステップ $S 3 1 2$ に戻ってループを繰り返す。そして、ループの繰り返しにより、閉曲線ナンバー k が 5 から 4、3、2、1 というように順次上がって、 $k=0$ になると、一番高い優先度になったと判断して図 12 のステップ $S 3 2 2$ に抜ける。このようにして、各閉曲線に対してカラー状態フラグ $Cflag[k]$ が反転することにより、色を塗る許可が与えられるとともに、カラー番号によって塗るべき色が指定され、その閉曲線の優先度が付与される。

【0053】図 11 に移り、閉曲線のラインに対して今度はドット毎に処理するプロセスを実行する。まず、ステップ $S 3 2 2$ では閉曲線ナンバー k を (n) に設定する。例えば、閉曲線が 6 個あるとすると、閉曲線ナンバー k を $[6]$ に設定する。閉曲線ナンバー k は値が小さい程、優先度が高くなるものであるから、ステップ $S 3 2 2$ では一番優先度の低い状態に設定されることになる。次いで、ステップ $S 3 2 4$ でカラー状態フラグ $Cflag[k]$ が $[-1]$ に等しくない状態であるか、すなわち閉曲線が色で塗りつぶされる状態になった否かを判別する。なお、この判別はフローチャートでは $Cflag[k] != -1$ として表され、「!」の符号は「C 言語」で否定 (not) を示すものである。ステップ $S 3 2 4$ の判別は優先度の低い閉曲線からカラー状態フラグ $Cflag[k]$ を判断することを意味している。

【0054】カラー状態フラグ $Cflag[k]$ が $[-1]$ に等しくない状態 ($TRUE$) であれば、ステップ $S 3 2 6$ に進んでカラーバッファ $Cbuf$ に閉曲線ナンバー k を入れる。これにより、今回指定された閉曲線ナンバー k の閉曲線に対して色が塗られることになる。次いで、

ステップ $S 3 2 8$ で閉曲線ナンバー k を $[1]$ だけデクリメントする。これにより、1つだけ優先度が高くなる。次いで、ステップ $S 3 3 0$ で閉曲線ナンバー k が $[0]$ 未満である（つまり、未だ一番高い優先度にはなっていない状態である）か否かを判別する。 $k=0$ でなければ（最初のルーチンでは優先度は最高になっていない）、 $TRUE$ に分岐してステップ $S 3 2 4$ に戻ってループを繰り返す。そして、ループの繰り返しにより、閉曲線ナンバー k が 6 から 4、3、2、1 というように順次上がって、 $k=0$ になると、一番高い優先度になったと判断してステップ $S 3 3 2$ に抜ける。一方、ステップ $S 3 2 4$ でカラー状態フラグ $Cflag[k]$ が $[-1]$ に等しい状態 ($FALSE$) であれば、ステップ $S 3 2 6$ をジャンプしてステップ $S 3 2 8$ に進む。したがって、このときはカラーバッファ $Cbuf$ に閉曲線ナンバー k が入れられず、今回指定された閉曲線ナンバー k の閉曲線に対して色が塗られないことになる。このようにして、優先度の低い閉曲線からカラーバッファ $Cbuf$ に閉曲線ナンバー k が入れられる。

20 【0055】ステップ $S 3 3 2$ ではラインカラー $lcolor$ が $[-1]$ に等しくない状態であるか、すなわち閉曲線のカラーが指定されたか否かを判別する。なお、この判別はフローチャートでは $lcolor != -1$ として表される。ラインカラー $lcolor$ が $[-1]$ に等しくない状態であれば（閉曲線のカラーが指定されていれば）、ステップ $S 3 3 4$ に進んでラインナンバー（ライン番号） $lnum$ がカラーバッファ $Cbuf$ より小さいか否かを判別する。カラーバッファ $Cbuf$ には優先度の高い曲線の色が入るから、ラインナンバー（ライン番号） $lnum$ がカラーバッファ $Cbuf$ よりも小さければ、 $TRUE$ に分岐してステップ $S 3 3 6$ に進み、ポインタ i, j によって指定されるラインおよびドットのカラー $color[i][j]$ （以下、ポインタ指定カラーという）を優先度の高いラインカラー $lcolor$ にセットし、その後、ステップ $S 3 3 8$ に進む。これにより、優先度の高い曲線の色がその閉曲線に塗られることになる。

30 【0056】一方、ステップ $S 3 3 2$ でラインカラー $lcolor$ が $[-1]$ に等しい状態（閉曲線のカラーが指定されていない状態）であれば、ステップ $S 3 4 0$ に分岐してポインタ指定カラー $color[i][j]$ をカラー状態フラグ $Cflag[Cbuf]$ にセットしてステップ $S 3 3 8$ に進む。カラー状態フラグ $Cflag[Cbuf]$ はバックグラウンド（背景）の色を塗る状態を示すフラグである。したがって、この場合にはその閉曲線にバックグラウンド（背景）の色が塗られる。また、ステップ $S 3 3 4$ でラインナンバー（ライン番号） $lnum$ がカラーバッファ $Cbuf$ 以上のときには、 $FALSE$ に分岐してステップ $S 3 4 0$ に進み、同様にポインタ指定カラー $color[i][j]$ をカラー状態フラグ $Cflag[Cbuf]$ にセットしてステップ $S 3 3 8$ に進む。したがって、この場合もそ

の閉曲線にバックグラウンド（背景）の色が塗られる。このようにして、今回のポインタ i 、 j によって指定されたラインおよびドットに対して優先度の高い閉曲線の色が塗られたり、あるいはバックグラウンド（背景）の色が塗られたりする。

【0057】次いで、ステップS338でポインタ j を [1] だけインクリメントして次のドットに移行する。次いで、ステップS342でインクリメント後のポインタ j が $ndot$ （例えば、1ラインの最大値で $ndot=256$ ドット）まで到達したか否かを判別し、到達していなければ図10のステップS308に戻って同様のルーチンを繰り返す。これにより、次のルーチンでは同じライン上の次のドットに対して同様の処理が行われる。そして、ステップS342でポインタ j が $ndot$ （例えば、256ドット）まで到達すると、1ラインの全てのドットについてカラー設定処理が終了したと判断してステップS344に抜ける。

【0058】ステップS344ではポインタ i を [1] だけインクリメントして次のラインに移行する。次いで、ステップS346でインクリメント後のポインタ i が $nline$ （例えば、1画面の最大値で $nline=525$ 本）まで到達したか否かを判別し、到達していなければ図10のステップS306に戻って再びドットを指定するポインタ j を [0] に戻して同様のルーチンを繰り返す。これにより、次のルーチンでは次のラインに移行して同様の処理が行われる。そして、ステップS346でポインタ i が $nline$ （例えば、525本）まで到達すると、全てのラインについてカラー設定処理が終了したと判断して本ルーチンを終了する。

【0059】このように、1画面をラインとドットに分けて、カラー設定処理が行われ、このとき描画する全ての閉曲線（例えば、似顔絵の各部位に対応する閉曲線）についてのフラグ設定を終了した後、全ての閉曲線のフラグを参考にしてカラー設定エリアにカラー番号が設定される。この場合、何れかの閉曲線の境界線が検出されたときには、その境界線がいま属している閉曲線よりも優先度の高い閉曲線の内部にあるときには、優先度の高い内部カラーをカラー設定エリアに設定するようにカラー状態フラグが制御される。したがって、常に優先度の高い閉曲線のカラーで塗りつぶされることになり、似顔絵が簡単なデータ構成で作成される。具体的には、例えば図4に示すように似顔絵の各部位（前髪、髪型、髪の光沢、顔の輪郭、顔のパーツ、首）に対応する閉曲線を作成するとともに、各部位のカラー設定エリアにカラー番号を設定していき、画面に表示するときには、優先度の高い各部位のカラーでその閉曲線を塗りつぶす処理が行われる。これにより、似顔絵の完成画面が表示部5に表示される。

【0060】図形変形処理の原理説明

次に、少量のデータによって同じ人物で異なる表情を作

成するために用いる図形変形処理について、その原理を説明する。この原理は、前述した変形データの変形情報に応じて似顔絵の表情を変形させる場合に適用される。ここでは、図13に示すように、例えば長方形100を複数の三角形に分割して、その三角形の変形に伴って図形を変形させる場合の原理を説明する。長方形100内の三角形の分割方法は図13に示す通りであり、8個の三角形に分割するが、このとき何れの三角形も位置変換のための演算を容易するため、三角形が全て直角三角形になるように分割する。まず、変形の条件としては長方形100の中心点 TP を分割中心として移動させないことで、データ数を少なくすることとし、直角三角形の変形には図14に示すような位置変換1を適用し、直角三角形の変形には図15に示すような位置変換2を適用する。

【0061】A. 直角三角形のフレーミングによる位置変換1

図14に示すものは直角三角形のフレーミングによる位置変換1の例であり、 T_1 、 T_2 、 T_3 を頂点とする直角三角形101を、頂点 T_3 の位置を変えないで、 T_1' 、 T_2' 、 T_3 を頂点とする三角形102に変形するものである。このとき、直角三角形101から三角形102への位置変換では、以下に述べる演算が行われる。直角三角形101の内部の特定の点 P の x 座標は、各辺上の点 V_1 、 V_2 に対応させると、

$$V_1x = V_2x = Px$$

で表される。また、特定点 P と頂点 T_1 との相対位置 vr の関係は

$$vr = V_1x / T_1x = Px / T_1x$$

となる。

【0062】ここで、辺上の点 V_1 の y 座標は $V_1y = 0$ である。一方、直角三角形101の各辺上の点 V_1 、 V_2 を特定点 P と頂点 T_2 との相対位置 vr の関係に対応させると、

$$V_2y = T_2y \cdot vr$$

で表される。したがって、特定点 P の相対位置 pr は、

$$pr = (V_2y - V_1y) / (Py - V_1y)$$

$$= V_2y / Py$$

$$= T_2y \cdot Px / T_2x \cdot Py$$

で表される。また、変形後の三角形102の各辺上の点 V_1' 、 V_2' の座標は次式によって表すことができる。

$$V_1'x = T_1'x \cdot vr$$

$$V_1'y = T_1'y \cdot vr$$

$$V_2'x = T_2'x \cdot vr$$

$$V_2'y = T_2'y \cdot vr$$

【0063】以上のデータから、変形後の三角形102の特定点 P' の x 座標は、次のような演算処理で算出できる。まず、

$$(V_2'x - V_1'x) : (P'x - V_1'x)$$

$$= (V_2x - V_1x) : (Px - V_1x)$$

という関係が成り立つので、

$(V_2' x - V_1' x) / (P' x - V_1' x) = p r$ という関係が成立する。したがって、この $p r$ を用いて特定点 P' の x 座標を求めると、以下ようになる。

$$P' x = \{ (V_2' x - V_1' x) / P r \} + V_1' x = (T_2' x - T_1' x) \times (v r / P r) + T_1' x \cdot v r = (T_2' x - T_1' x) \times \{ (T_1 x \cdot P y) / (T_2 y \cdot P x) \} \times (P x / T_1 x) + T_1' x \cdot (P x / T_1 x) = (T_1' x / T_1 x) \cdot P x + \{ (T_2' x - T_1' x) / T_2 y \} \cdot P y$$

【0064】同様に、変形後の三角形102の特定点 P' の y 座標は、次のような演算処理で算出できる。まず、

$$(V_2' y - V_1' y) : (P' y - V_1' y) = (V_2 y - V_1 y) : (P y - V_1 y)$$

という関係が成り立つので、

$(V_2' y - V_1' y) / (P' y - V_1' y) = p r$ という関係が成立する。したがって、この $p r$ を用いて特定点 P' の y 座標を求めると、以下ようになる。

$$P' y = \{ (V_2' y - V_1' y) / P r \} + V_1' y = (T_2' y - T_1' y) \times (v r / P r) + T_1' y \cdot v r = (T_2' y - T_1' y) \times \{ (T_1 x \cdot P y) / (T_2 y \cdot P x) \} \times (P x / T_1 x) + T_1' y \cdot (P x / T_1 x) = (T_1' y / T_1 x) \cdot P x + \{ (T_2' y - T_1' y) / T_2 y \} \cdot P y$$

【0065】このように、 T_1 、 T_2 、 T_3 を頂点とする直角三角形101を、頂点 T_3 の位置を変えないで、 T_1' 、 T_2' 、 T_3 を頂点とする三角形102に変形する際、変形前の直角三角形101の内部の特定点 P の x 座標および y 座標は、各辺上の点 V_1 、 V_2 に対応させて変形後の特定点 P' として算出することができる。これは、言換えれば特定点 P を似顔絵を構成する変形前の各ドットに当てはめた場合に、変形後の各ドットは特定点 P' として求めることが可能なことを意味している。しかも、このとき頂点 T_3 の位置は変えないので、変形に必要なデータが少なく済む。したがって、このような変形処理により、同じ人物として同一性を確保しながら似顔絵の表情を変化させることが可能になる。

【0066】B. 直角三角形のフレーミングによる位置変換2

$$P' x = \{ (V_2' x - V_1' x) / P r \} + V_1' x = (T_2' x - T_1' x) \times (v r / P r) + T_1' x \cdot v r = (T_2' x - T_1' x) \times \{ (T_1 y \cdot P x) / (T_2 x \cdot P y) \} \times (P y / T_1 y) + T_1' x \cdot (P y / T_1 y) = (T_2' x - T_1' x) / T_2 x \cdot P x + (T_1' x / T_1 y) \cdot P y$$

【0069】同様に、変形後の三角形112の特定点 P' の y 座標は、次のような演算処理で算出できる。

$$(V_2' y - V_1' y) / (P' y - V_1' y) = p r$$

$$P' y = \{ (V_2' y - V_1' y) / P r \} + V_1' y = (T_2' y - T_1' y) \times (v r / P r) + T_1' y \cdot v r$$

図15に示すものは直角三角形のフレーミングによる位置変換2の例であり、 T_1 、 T_2 、 T_3 を頂点とする直角三角形111を、頂点 T_3 の位置を変えないで、 T_1' 、 T_2' 、 T_3 を頂点とする三角形112に変形するものである。このとき、直角三角形111から三角形112への位置変換では、以下に述べる演算が行われる。直角三角形111の内部の特定点 P の y 座標は、各辺上の点 V_1 、 V_2 に対応させると、

$$V_1 y = V_2 y = P y$$

10 で表される。また、特定点 P と頂点 T_1 との相対位置 $v r$ の関係は

$$v r = V_1 y / T_1 y = P y / T_1 y$$

となる。

【0067】ここで、辺上の点 V_1 の x 座標は $V_1 x = 0$ である。一方、直角三角形111の各辺上の点 V_1 、 V_2 を特定点 P と頂点 T_2 との相対位置 $v r$ の関係に対応させると、

$$V_2 x = T_2 x \cdot v r$$

で表される。したがって、特定点 P の相対位置 $p r$ は、

$$20 \quad p r = (V_2 x - V_1 x) / (P x - V_1 x) = V_2 x / P x = T_2 x \cdot P y / T_1 y \cdot P x$$

で表される。また、変形後の三角形112の各辺上の点 V_1' 、 V_2' の座標は次式によって表すことができる。

$$V_1' x = T_1' x \cdot v r$$

$$V_1' y = T_1' y \cdot v r$$

$$V_2' x = T_2' x \cdot v r$$

$$V_2' y = T_2' y \cdot v r$$

30 【0068】以上のデータから、変形後の三角形112の特定点 P' の x 座標は、次のような演算処理で算出できる。まず、

$$(V_2' x - V_1' x) : (P' x - V_1' x) = (V_2 x - V_1 x) : (P x - V_1 x)$$

という関係が成り立つので、

$$(V_2' x - V_1' x) / (P' x - V_1' x) = (V_2 x - V_1 x) / (P x - V_1 x) = V_2 x / P x$$

= $p r$ という関係が成立する。したがって、この $p r$ を用いて特定点 P' の x 座標を求めると、以下になる。

$$\begin{aligned}
 &= (T_2' y - T_1' y) \times \{ (T_1 y \cdot P x) / (T_2 x \cdot P y) \} \\
 &\quad \times (P y / T_1 y) + T_1' y \cdot (P y / T_1 y) \\
 &= (T_2' y - T_1' y) / T_2 x \cdot P x + (T_1' y / T_1 y) \cdot P y
 \end{aligned}$$

【0070】このように、 T_1 、 T_2 、 T_3 を頂点とする直角三角形111を、頂点 T_3 の位置を変えないで、 T_1' 、 T_2' 、 T_3 を頂点とする三角形112に変形する際、変形前の直角三角形111の内部の特定点Pのx座標およびy座標は、各辺上の点 V_1 、 V_2 に対応させて変形後の特定点P'として算出することができる。これは、言換えれば特定点Pを似顔絵を構成する変形前の各ドットに当てはめた場合に、変形後の各ドットは特定点P'として求めることが可能なことを意味している。しかも、このとき頂点 T_3 の位置は変えないので、変形に必要なデータが少なく済む。したがって、このような変形処理により、同じ人物として同一性を確保しながら似顔絵の表情を変化させることが可能になる。

【0071】以上のように、長方形100の内部を複数の（本実施例では8個の）直角三角形 ～ に分割し、その三角形の変形に伴って図形を変形させることにより、変形に必要なデータとして長方形100の中心点T

P（分割中心のデータ）、変形させるパーツのデータ（部位のデータ）、長方形100の元のデータ、変形後のデータだけよいという少ないデータで容易に似顔絵の表情を変化させることができる。本実施例では、このような変形方法を用いて似顔絵の基本画像が上述した各プログラムの実行によって変形され、似顔絵の表情を変化させることが行われる。具体的には、例えば図16に一例を示すようにメインプログラムで似顔絵のカラーの基本画像201が作成された後、そのカラー画像を見ながら選択スイッチを操作して変形データの選択番号を変更すると、選択番号に対応した変形情報（すなわち、変形データ）がROM2から読み出され、基本画像201の一部を形成する閉曲線（例えば、顔のパーツを形成する閉曲線）が変形して、結局、図16の変形画像202のようになり、似顔絵の表情が変化（例えば、驚いたような表情に変化）したものが表示部5に表示される。当然のことながら、選択番号を変え、似顔絵201を同一人物で別の異なる表情（例えば、笑ったような表情）に変化させることができる。

【0072】以上のことから、似顔絵の画像をカラー表示させてその表情を変化させる場合でも、従来のようにピクセル毎にカラーデータを持つ必要がなく、単に閉曲線データとカラーデータとを持てば画像をカラー表示することが可能になるとともに、さらに従来のように似顔絵等の全部又は一部のデータを表情の変化数分だけ持つ必要がなく、単に閉曲線データ、カラーデータおよび似顔絵の表情を変化させる変形データ（表情の変化に対応する変形データ）を持てば、似顔絵の基本画像の表情を変化させることができる。すなわち、同じ人物の似顔絵で別の表情を少ないデータでさせることができる。その

結果、データ量が少なく、メモリ装置の容量を少なくしつつ、かつ低コストで似顔絵の画像をカラー表示し、かつその表情を変化させることができる。

【0073】次に、本発明の第2実施例について図17～図20を参照して説明する。本実施例は作成した基本画像の一部を修正枠で指定し、指定した基本画像の一部を変形データに応じて変形し、変形した基本画像の閉曲線の境界および内部を塗りつぶして、カラーの変形画像を作成し、表情を変化させるものである。本実施例のハード的なブロック構成は前記実施例の図1に示すものと同様であるが、スイッチ部4の中に修正スイッチ、修正スタートスイッチおよび塗りつぶしスイッチが含まれる点が異なる。修正スイッチ（修正枠指定手段）は似顔絵の一部を修正枠で指定するときに操作されるもので、修正スタートスイッチは修正枠で指定した部分の修正を開始するときに操作されるもので、さらに塗りつぶしスイッチは色を省略した似顔絵の基本画像に対して色で塗りつぶす処理を開始するときに操作されるものである。なお、CPU1、ROM2およびRAM3は全体として閉曲線作成手段、画像変形手段、画像作成手段を構成する。以下、異なるプログラム部分について説明する。

メインプログラム

図17は似顔絵作成処理のメインプログラムを示すフローチャートである。このプログラムがスタートすると、まずステップS400でイニシャライズ（初期化处理）を行う。これにより、例えばCPU1内の各種レジスタのイニシャライズ、RAM3のワークエリアのクリア、サブルーチンのイニシャライズ、フラグのリセット等が行われる。

【0074】次いで、ステップS402でスタートスイッチがオンしているか否かを判別する。

A. スタートスイッチがオンしているとき

スタートスイッチがオンしていれば、ステップS404で表示部5の表示をクリアする。ここで最初のルーチンでは、例えば初期画面を表示させてもよいし、あるいは何等かの操作画面を表示してもよいし、あるいは何も表示させなくてもよいが、何れにしても、ステップS404で一旦表示がクリアされる。また、例えば既に似顔絵が表示されている場合には、以前の似顔絵がクリアされる。

【0075】ここで、ROM2には図18に示すように基準となる似顔絵（基本画像）を作成するための似顔絵データおよびその似顔絵の表情を変化させるための複数の変形データが格納されている。なお、複数の変形データは（AD=1）～（AD=n）というアドレスで示すエリアに格納されている。似顔絵データは似顔絵の輪郭や顔の各部位を表す閉曲線作成データA～Fと、閉曲線

の色を指定する閉曲線カラーデータA～Fとに区分されて所定のエリアに格納されている。閉曲線カラーデータA～Fは、閉曲線作成データA～Fのそれぞれの色（閉曲線の境界および内部を塗りつぶす色）を指定するものである。また、(AD=1)～(AD=n)というアドレスで示すエリアに格納されている複数の変形データは基準となる似顔絵の表情をどのように変化させるかを示すデータであり、それぞれ予め設定されている。例えば、変形データ(AD=1)は怒った表情、変形データ(AD=2)は笑った表情という具合に、似顔絵の表情を変化させるものであり、選択スイッチによって選択が可能である。なお、この場合に似顔絵をどのような表情で表示させるかにより、変形データを自由に設定することが可能であるとともに、変形データの内容（変形情報）も自由に設定可能である。

【0076】さらに、本実施例では基本画像の一部分を修正するために修正枠のデータが複数格納されている。なお、複数の修正枠のデータは(AD=0)～(AD=n)というアドレスで示すエリアに格納されている。複数の修正枠のデータは基準となる似顔絵の一部分をどのように変化させるかを示すデータであり、それぞれ予め設定されている。例えば、修正枠のデータ(AD=0)は普通の表情、修正枠のデータ(AD=1)は怒った表情、修正枠のデータ(AD=2)は笑った表情という具合に、似顔絵の一部の表情を変化させるものであり、修正スイッチによって選択が可能である。なお、この場合に似顔絵をどのような表情で表示させるかにより、修正枠のデータを自由に設定することが可能であるとともに、修正枠のデータの内容（修正情報）も自由に設定可能である。

【0077】再びプログラムに戻り、次いで、ステップS406で基本となる似顔絵を作成するために、閉曲線作成データA～Fおよび閉曲線カラーデータA～FをROM2から読み出して、RAM3にロードする。ここで、RAM3には図19に示すような各種のワークエリアが設けられており、ROM2から読み出されたデータは対応するエリアにロードされる。RAM3のワークエリアを説明すると、以下のようにになっている。閉曲線A作成データ～閉曲線F作成データ、閉曲線Aカラーデータ～閉曲線Fカラーデータ、カラー状態フラグ(A)～カラー状態フラグ(F)、バックグラウンドカラー番号、選択番号をそれぞれ格納するエリアは前記実施例と同様である。その他に、修正枠のデータを格納するエリアおよび修正枠によって表情を変化させる変形情報を格納するエリアが設けられている。さらに、作成した似顔絵の各部位に対応する閉曲線を格納するエリア11～16があり、例えば、エリア11は前髪を格納するエリア、エリア12は髪型を格納するエリア、エリア13は髪における光沢部分を格納するエリア、エリア14は顔の輪郭を格納するエリア、エリア15は顔のパーツを格

納するエリア、エリア16は首を格納するエリアである。

【0078】再びフローチャートの説明に戻り、ステップS406を経ると、続くステップS408に進み、ロードした閉曲線作成データA～Fに基づいて閉曲線A～Fをそれぞれ作成する処理（詳細は前述したサブルーチンと同様）を行うとともに、ステップS410では表示処理を行う。これにより、色を省略した似顔絵の基本画像が閉曲線によって作成され、表示部5に表示される。なお、この時点では閉曲線A～Fを塗りつぶす処理は行われていない。次いで、ステップS412でアドレスポインタADを[0]にクリアする。アドレスポインタADはROM2に格納されている修正枠を指定していくものである。また、このステップでスタートフラグSFを[1]にセットする。スタートフラグSFは、イニシャライズ処理および後述のステップS452で[0]にクリアされ、スタートスイッチのオン操作により基本画像（ここでは色を省略した似顔絵の基本画像）が作成されたときに、このステップで[1]にセットされる。そして、SF=[1]になると、基本画像の表情を変化させることが可能になる。

【0079】ステップS412でSF=[1]になると、ステップS414に進む。一方、上記ステップS402でスタートスイッチがオンしていなければ、ステップS402でNOに分岐しステップS414に直接ジャンプする。したがって、このときは基本画像が作成されない。ステップS414では修正スイッチがオンしているか否かを判別する。修正スイッチは基本画像の一部を修正枠で指定して表情を変化させる場合に、操作されるものであるから、修正スイッチがオンしていれば後述のステップで修正枠を読み出すことになる。また、修正スイッチがオンしていなければ、NOに分岐してステップS444にジャンプする。最初に、修正スイッチがオンしていないケースから説明する。

【0080】A. 修正スイッチがオンしていないとき
ステップS414からNOに分岐してステップS444にジャンプし、ステップS444で塗りつぶしスイッチがオンしているか否かを判別する。塗りつぶしスイッチは、色を省略した似顔絵の基本画像に対してカラー処理（すなわち、色で塗りつぶす処理）を開始するときに操作されるものである。塗りつぶしスイッチがオンしていると、ステップS446に進んで修正枠をクリアし（例えば、後述の図20に示すような修正枠311の表示をクリア）、ステップS448で塗りつぶす処理（詳細は前述したサブルーチンと同様）を行う。これにより、表情はそのままだの色無しの基本画像の各閉曲線に対して色を付ける処理が行われ、カラーの似顔絵画像が作成されて表示部5に表示される。

【0081】次いで、ステップS450でストップスイッチがオンしているか否かを判別する。ストップスイ

チは似顔絵の表示を停止させるときに操作されるものであるから、ストップスイッチがオンしていると、ステップS452に進んでスタートフラグSFを[0]にクリアするとともに、アドレスポインタADを同様に[0]に戻す。これにより、次の修正枠処理に備えられる。次いで、ステップS402に戻ってスタートスイッチの操作を判別する。また、ステップS450でストップスイッチがオンしていなければ、ステップS452をジャンプしてステップS402に戻り、同様にスタートスイッチの操作を判別する。したがって、このときは次の修正枠処理に備えることは行われず、そのままの表情の似顔絵の表示が継続する。例えば、同じ表情の似顔絵を表示しておくような場合である。

【0082】B. 修正スイッチがオンしたとき
ステップS414からYESに分岐し、続くステップS416でスタートフラグSFが[1]であるか否かを判別する。スタートフラグSFが[0]のときはステップS444にジャンプする。したがって、このときは修正スイッチをオン操作しても似顔絵の変形は行われない。ステップS416でスタートフラグSFが[1]のときはステップS418に進んでAD番目の修正枠のデータをROM2から読み出し、RAM3にロードする。例えば、最初のルーチンではAD=0であるから、図19に示すように通常の四角型の修正枠が読み出されてRAM3にロードされる。また、2回目以降のルーチンでAD=1であれば、図19に示すように星型の修正枠が読み出されてRAM3にロードされる。

【0083】次いで、ステップS420で読み出した修正枠を表示画面に重ねて表示する。例えば、図20に示すように基本画像301に対して通常の四角型の修正枠311を重ねて表示する。これにより、似顔絵を変形する部分が修正枠311によって明らかになる。次いで、ステップS422で似顔絵の表情を選択するときには操作される選択スイッチがオンしたか否かを判別する。選択スイッチがオンしていると、ステップS424に進んで修正枠の表示をクリアする。これにより、表示部5の表示から修正枠が消える。次いで、ステップS426で修正枠を指定するアドレスポインタADを[1]だけ進める。したがって、次の修正枠に進むことになる。次いで、ステップS428でAD番目の修正枠を表示する。例えば、アドレスポインタADを[1]だけ進めてAD=3になった場合には、図19に示すようにほぼ菱形の修正枠が読み出されてRAM3にロードされることとなる。

【0084】次いで、ステップS430でアドレスポインタADが(END+1)という最終番地になったか否か、つまり全ての修正枠を表示したか否かを判別する。NOのときはステップS434にジャンプする。したがって、このときは次回以降のルーチンで他の修正枠を読み出して基本画像の表情を変化させることになる。一

方、アドレスポインタADが最終番地(END+1)になったときはステップS430に進んでアドレスポインタADを再び[0]にクリアする。したがって、このときは次回以降のルーチンで、最初の修正枠から読み出しが始まることになる。ステップS434では修正スタートスイッチがオンしているか否かを判別する。修正スタートスイッチがオンしていればステップS436に進んで今回指定した修正枠に対応するAD番目の変形情報(すなわち、変形データ)をROM2から読み出し、RAM3にロードする。例えば、AD=1であれば、変形データ(AD=1)という情報が読み出される。

【0085】次いで、ステップS438で一旦閉曲線表示の画像をクリアする。これにより、いままで表示されていた色無しで閉曲線からなる似顔絵が消える。次いで、ステップS440で変形情報に基づき閉曲線を変形する。例えば、AD=1であれば、変形データ(AD=1)という情報に基づいて基本画像が変形されることになる。次いで、ステップS442で変形後の閉曲線を表示する。これにより、基本画像を変形した似顔絵が閉曲線によって表示される。次いで、ステップS444に進む。一方、ステップS434で修正スタートスイッチがオンしていなければ、ステップS436～ステップS442をジャンプしてステップS444に進む。したがって、このときは基本画像の変形は行われない。ステップS444以降の処理は前述した通りであり、塗りつぶしスイッチがオンしているか否かを判断して塗りつぶし処理が行われる。これにより、修正枠をクリアして基本画像が対応するカラーで塗りつぶされる。すなわち、表情を変形した色無しの基本画像の各閉曲線に対して色を付ける処理が行われ、表情の変化したカラーの似顔絵画像が作成されて表示部5に表示される。

【0086】図20は画像変形の一例を示す図である。最初は基本画像301が表示され、その基本画像301に対して通常の四角型の修正枠311を重ねて表示する。これにより、表情を変えたい部分が明らかになる。また、他の表情にするために、基本画像301に対して別の修正枠312を指定して修正スタートスイッチを押すと、修正枠312で指定された変形データに従って基本画像301の対応する閉曲線が変形し、変形画像302のように別の表情をした似顔絵となる。なお、この段階では色はない。次いで、カラーを付けると、カラーの変形画像303のようになり、カラーで表情の変化した似顔絵が得られる。この例は、通常の四角型の修正枠311を他の形状の修正枠312に変形したものである。

【0087】このように、本実施例では作成した基本画像の一部を修正枠で指定し、基本画像の一部を修正枠に対応する変形データに応じて変形し、変形した基本画像の閉曲線の境界および内部を塗りつぶして、カラーの変形画像を作成し、表情を変化させる処理が行われる。したがって、基本画像を見ながら、その一部を修正枠で自

由に指定して表情を変化させることができる（つまり、修正箇所を自由に選べる）とともに、修正の指定箇所が同じであっても、別の修正枠を選択することにより様々な表情に変化させることができるという効果がある。したがって、オペレータによる表情変化の自由度が増す。

【0088】上記各実施例では閉曲線を作成する場合の曲線パラメータとしてベジェ曲線のパラメータを使用しているが、これに限るものではなく、例えばBスプライン曲線等の任意の曲線を用いることも可能である。また、この他に放物線、双曲線、三角関数等のようなパラメータを適宜使用することも可能である。この場合、作成する閉曲線の形状に応じて放物線、双曲線、三角関数等の適切な数式を用いればよい。このようにすると、その時々に応じた適切な閉曲線を作成することができる。また、表示するカラー画像は似顔絵やアニメーション画像に限らず、ゲーム等で使用する各種の画像、キャラクターや背景データであってもよい。さらに、本発明はコンピュータ画面上あるいはテレビ画面上において、似顔絵やアニメーション画像の表情変化をカラー表示させる場合に限らず、他の分野、他の画像の表情変化にも適用できる。

【0089】

【発明の効果】本発明によれば、カラーの画像（例えば、似顔絵あるいはアニメーション画像）を作成してその表情を変化させる場合、まず色を省略した複数の閉曲線で基本画像を作成し、次いで、作成した基本画像を変形データに応じて変形して変形画像を作成し、この変形画像を構成する各閉曲線の境界および内部を優先順位に従って所定の色で塗りつぶしてカラーの変形画像を作成し、これによりカラーの基本画像の表情を変化させているので、以下の効果を得ることができる。

例えば、似顔絵の動画をカラー表示させてその表情を変化させる場合でも、従来のようにピクセル毎にカラーデータを持つ必要がなく、単に閉曲線データとカラーデータとを持てば画像をカラー表示することができるとともに、従来のように似顔絵等の全部又は一部のデータを表情の変化数分だけ持つ必要がなく、単に閉曲線データ、カラーデータおよび似顔絵の表情を変化させる変形データ（表情の変化に対応する変形データ）を持てば、似顔絵の基本画像の表情を変化させることができる。すなわち、同じ人物の似顔絵で別の表情を少ないデータで作成することができる。

その結果、データ量が少なく済み、メモリ装置の容量を少なくしつつ、かつ低コストで似顔絵やアニメーション等の画像をカラー表示し、かつその表情を変化させることができる。

【0090】 また、他の請求項記載の発明によれば、作成した基本画像の一部を修正枠で指定し、修正枠で指定した基本画像の一部を変形データに応じて変形し、変形した基本画像の閉曲線の境界および内部を塗りつぶし

て、カラーの変形画像を作成しているため、基本画像を見ながら、その一部を修正枠で自由に指定して表情を変化させることができる（つまり、修正箇所を自由に選べる）とともに、修正の指定箇所が同じであっても、別の修正枠を選択することにより様々な表情に変化させることができるという効果がある。したがって、オペレータによる表情変化の自由度が増すという利点がある。

【図面の簡単な説明】

【図1】本発明による似顔絵作成装置の第1実施例を示す構成図である。

【図2】同実施例の似顔絵作成処理のメインプログラムを示すフローチャートである。

【図3】同実施例のROMに格納されているデータを示す図である。

【図4】同実施例のRAMに格納されるデータを示す図である。

【図5】同実施例の閉曲線A～Fの作成処理のサブルーチンを示すフローチャートである。

【図6】同実施例のドット算出処理のサブルーチンを示すフローチャートである。

【図7】同実施例のドット算出処理のサブルーチンを示すフローチャートである。

【図8】同実施例のベジェ曲線を作成する場合のデータ計算処理のサブルーチンを示すフローチャートである。

【図9】同実施例のベジェ曲線を作成する場合のデータ計算処理のサブルーチンを示すフローチャートである。

【図10】同実施例のカラー設定処理のサブルーチンを示すフローチャートである。

【図11】同実施例のカラー設定処理のサブルーチンを示すフローチャートである。

【図12】同実施例のカラー状態フラグを説明する図である。

【図13】同実施例の図形変形処理の長方形を示す図である。

【図14】同実施例の図形変形処理を説明する図である。

【図15】同実施例の図形変形処理を説明する図である。

【図16】同実施例の似顔絵の表情変化の一例を示す図である。

【図17】本発明の第2実施例の似顔絵作成処理のメインプログラムを示すフローチャートである。

【図18】同実施例のROMに格納されているデータを示す図である。

【図19】同実施例のRAMに格納されるデータを示す図である。

【図20】同実施例の似顔絵の表情変化の一例を示す図である。

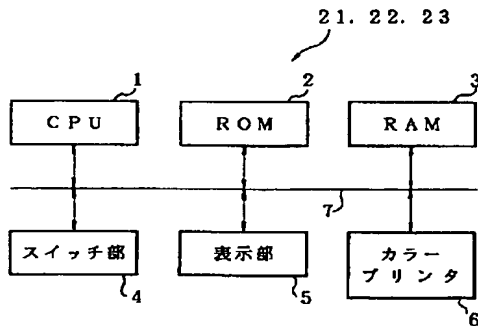
【符号の説明】

1 CPU

- 2 ROM (変形データ記憶手段)
3 RAM
4 スイッチ部
5 表示部 (表示手段)

- 6 印刷部 (印刷手段)
21 閉曲線作成手段
22 変形手段
23 画像作成手段

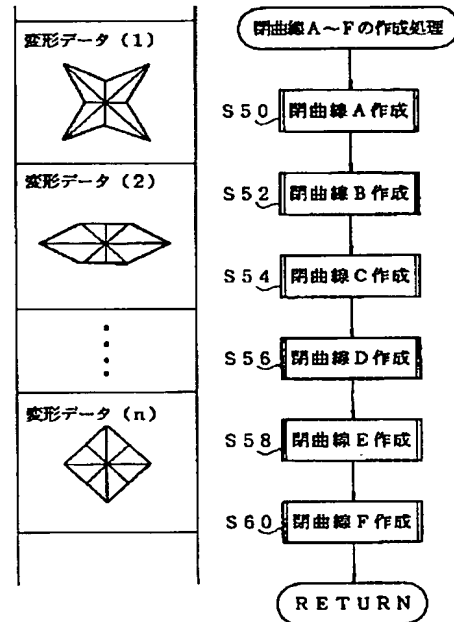
【図1】



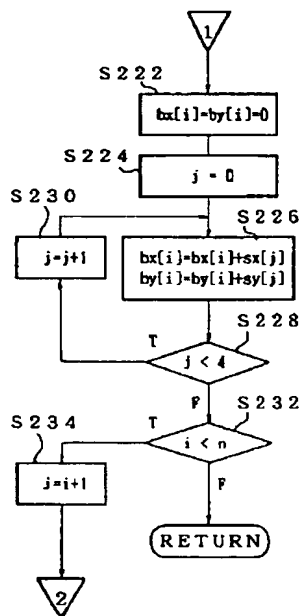
【図3】

閉曲線A作成データ
閉曲線B作成データ
閉曲線C作成データ
閉曲線D作成データ
閉曲線E作成データ
閉曲線F作成データ
閉曲線Aカラーデータ
閉曲線Bカラーデータ
閉曲線Cカラーデータ
閉曲線Dカラーデータ
閉曲線Eカラーデータ
閉曲線Fカラーデータ

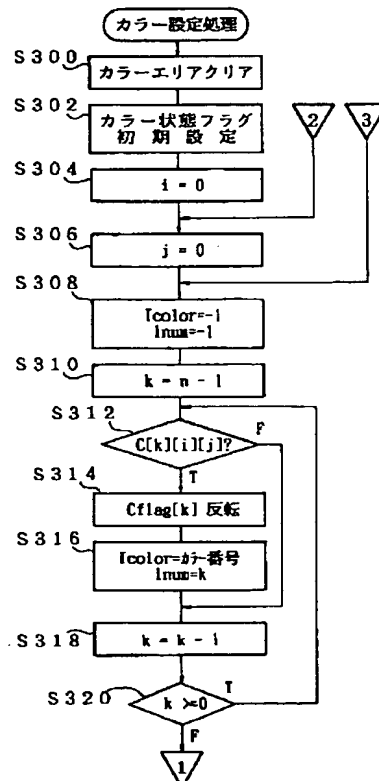
【図5】



【図9】



【図10】

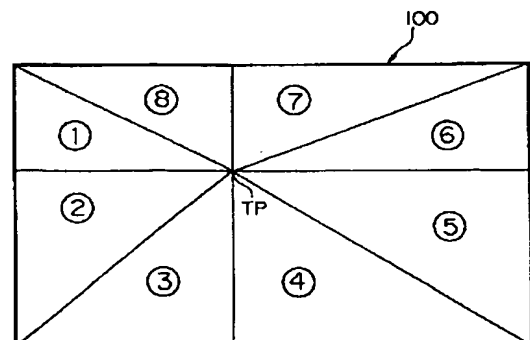


【図12】

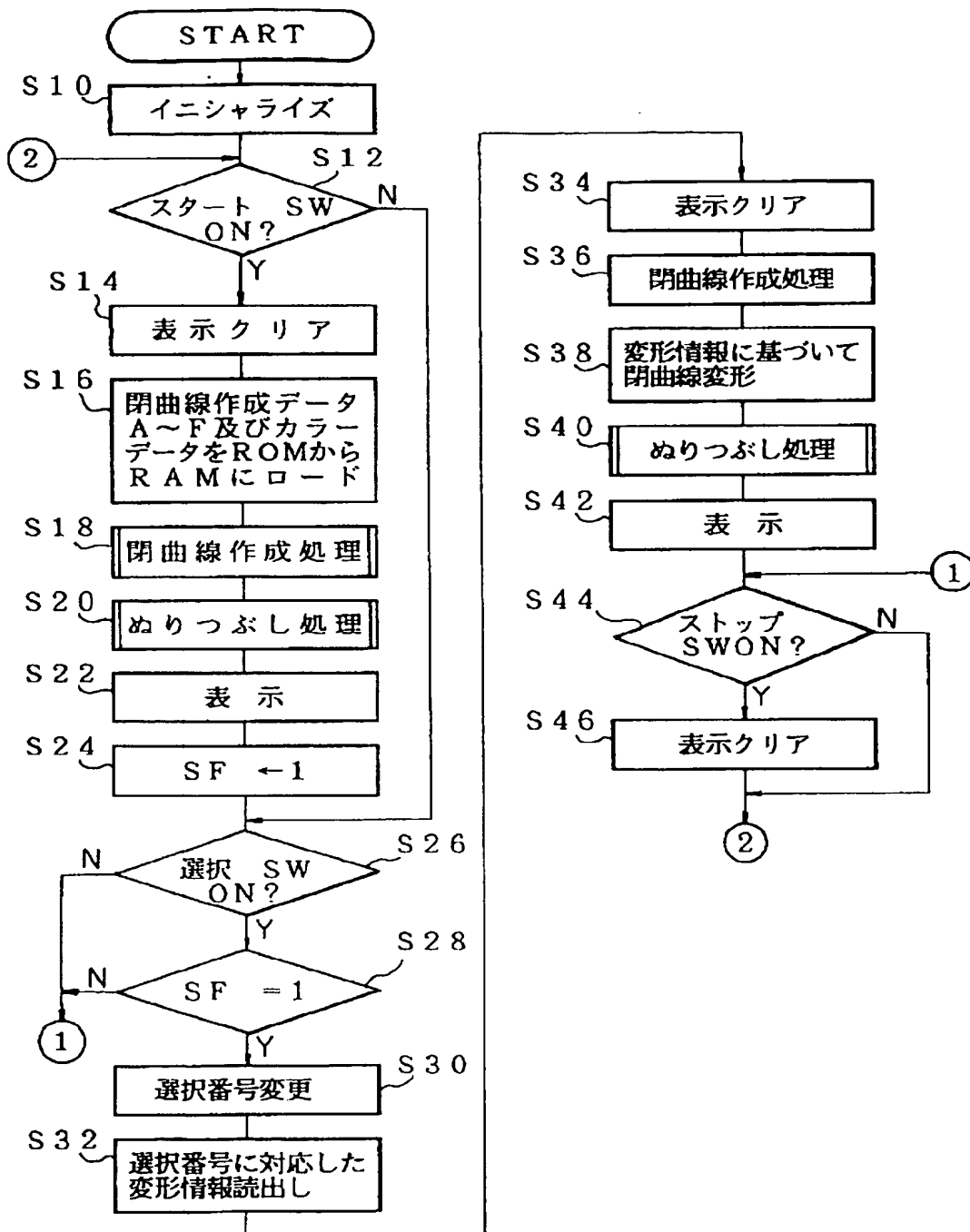
カラー状態フラグ (初期状態)

オブジェクト1のカラー	-1
オブジェクト2のカラー	-1
オブジェクト3のカラー	-1
...	...
オブジェクトnのカラー	-1
バックグラウンドのカラー	カラー番号

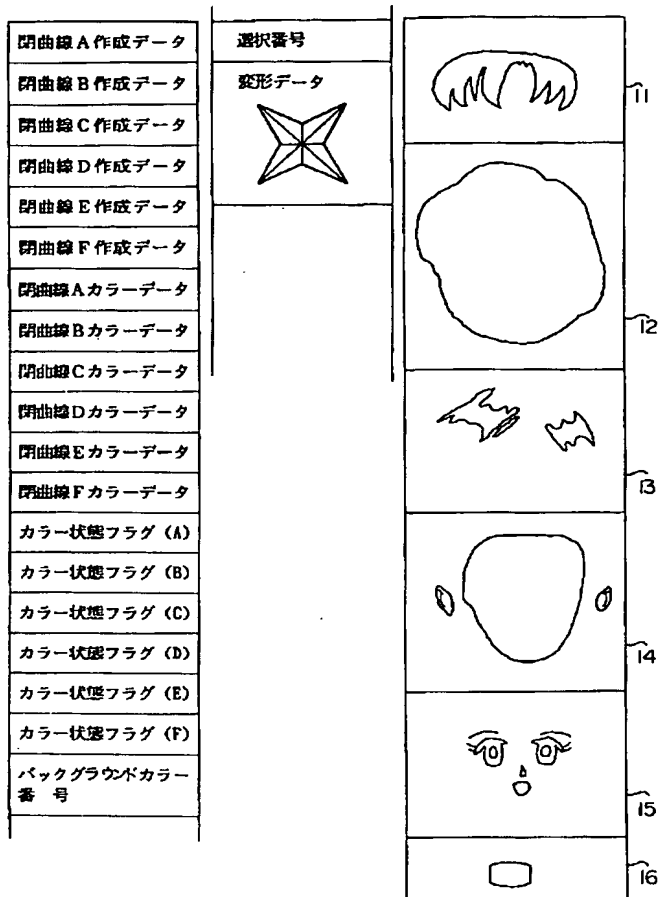
【図13】



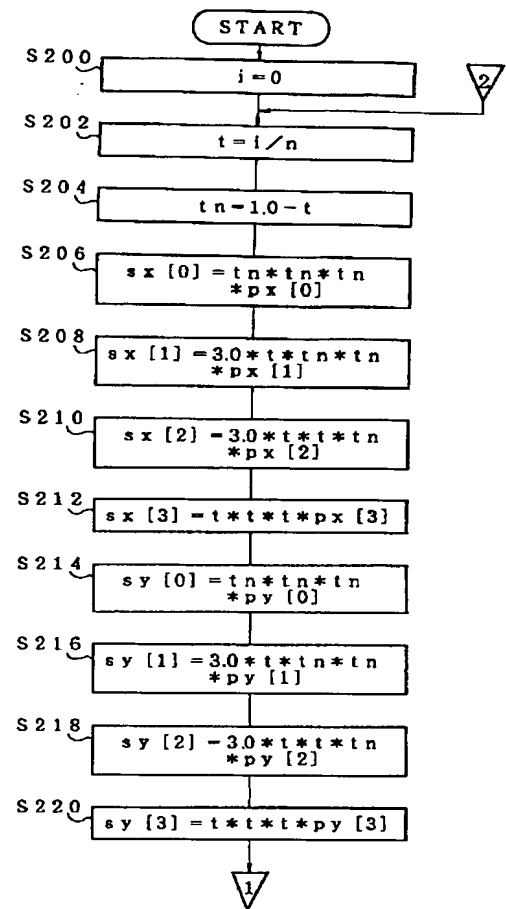
【図2】



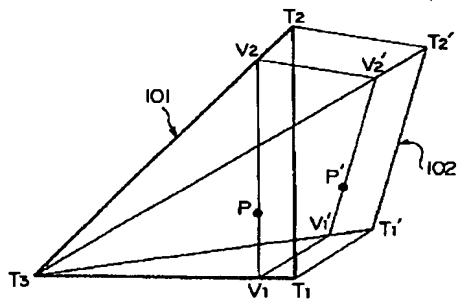
【図4】



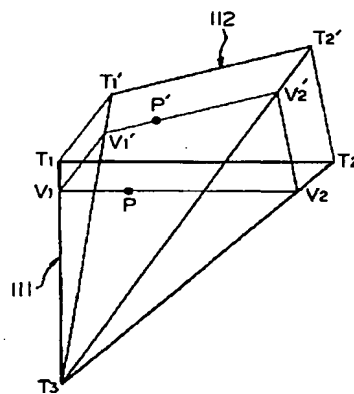
【図8】



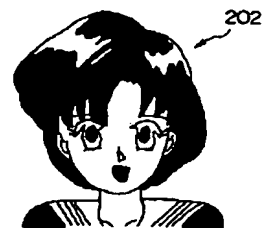
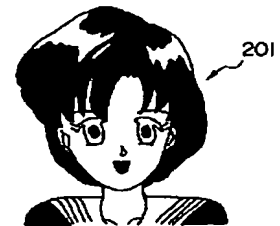
【図14】



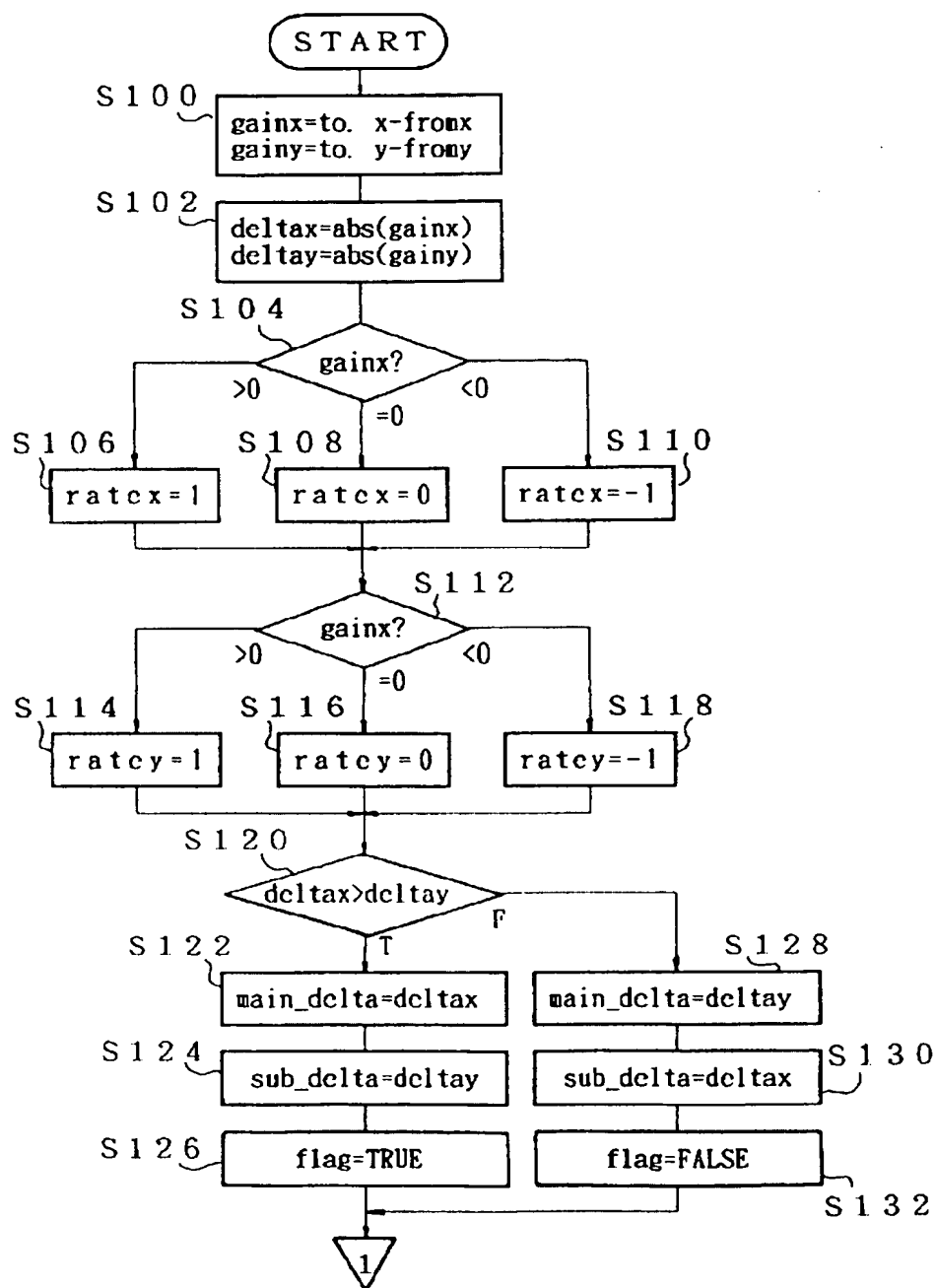
【図15】



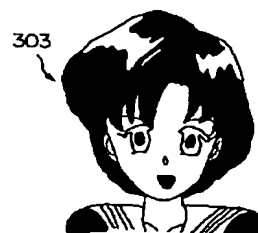
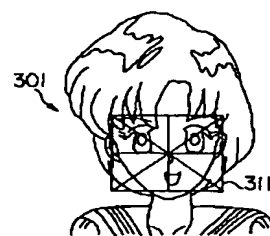
【図16】



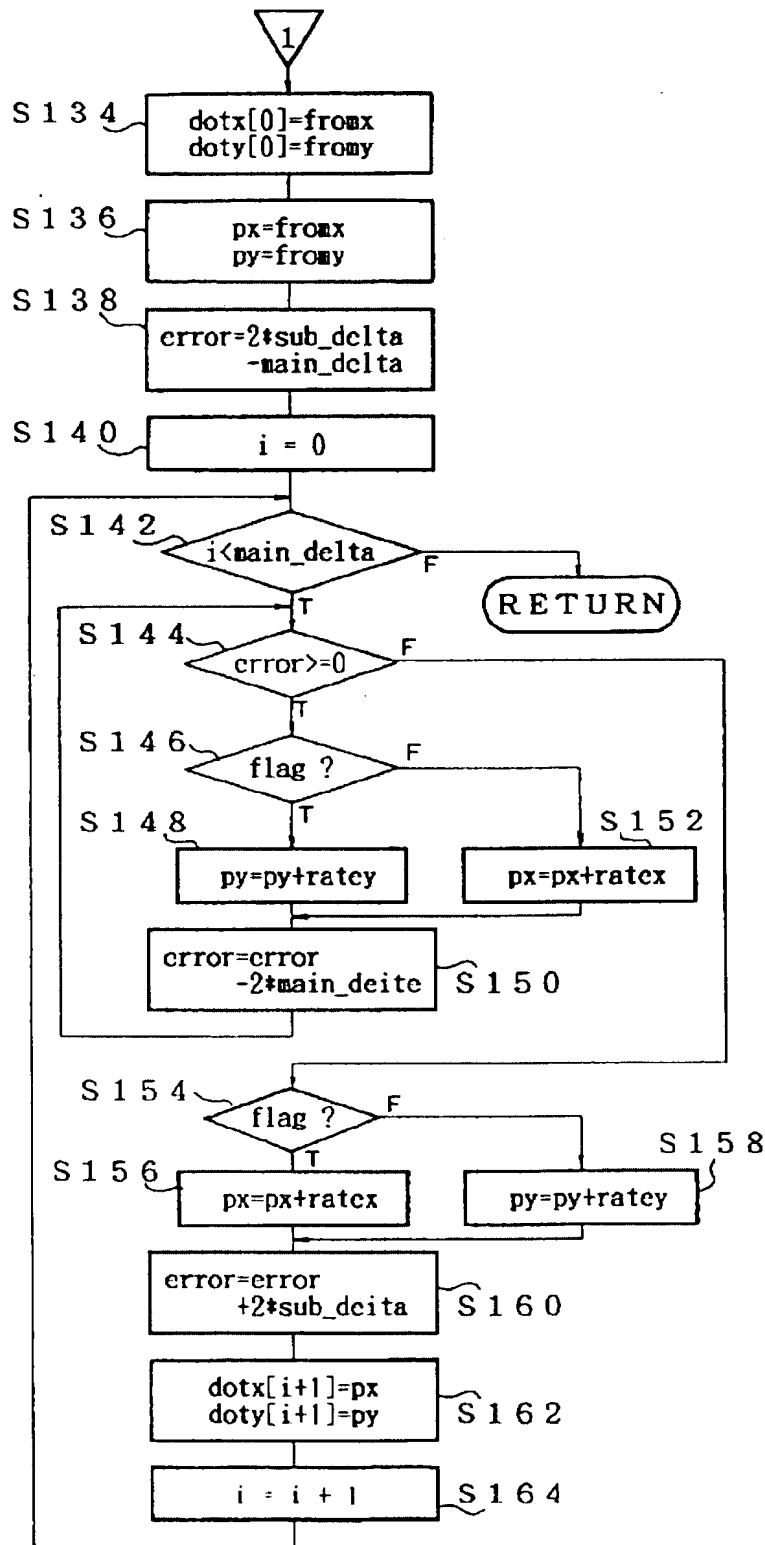
【図6】



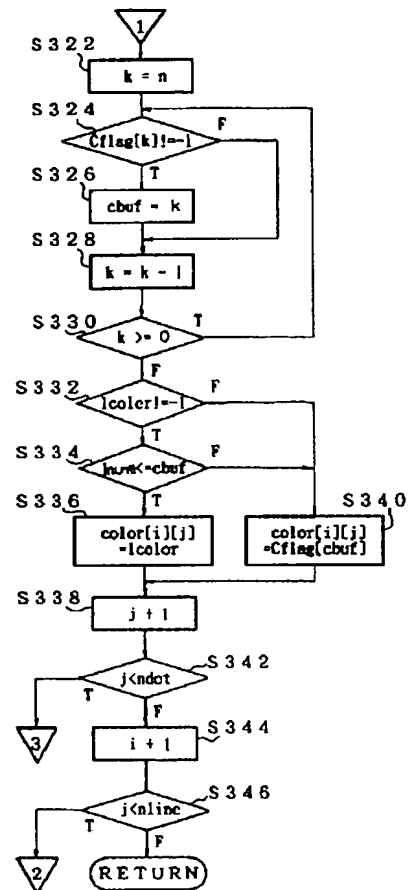
【図20】



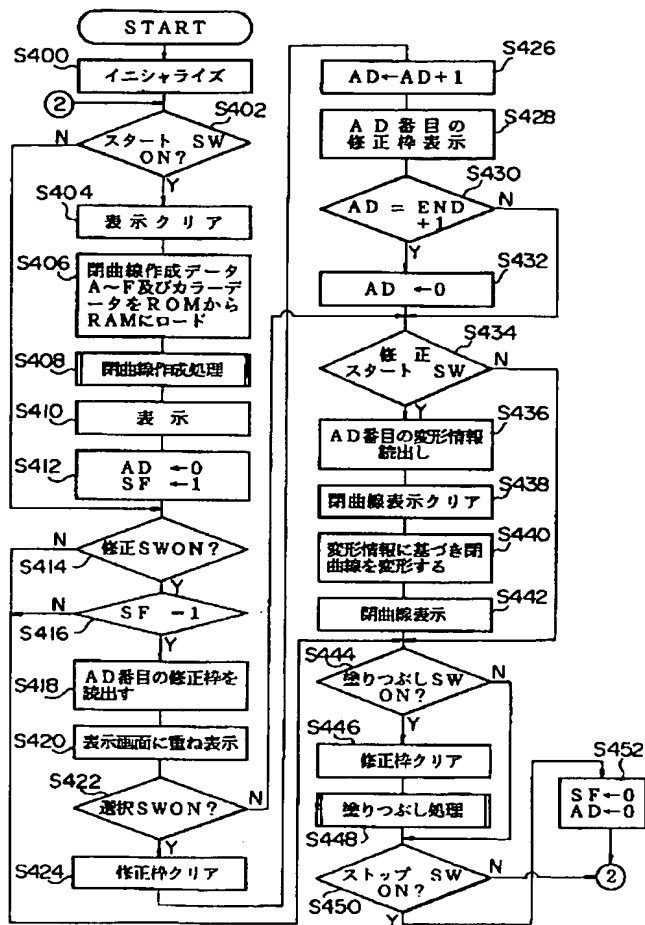
【図 7】



【図 11】



【図17】





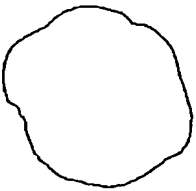

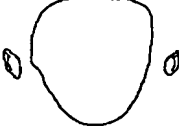
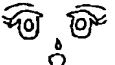

【図18】

閉曲線 A 作成データ
閉曲線 B 作成データ
閉曲線 C 作成データ
閉曲線 D 作成データ
閉曲線 E 作成データ
閉曲線 F 作成データ
閉曲線 A カラーデータ
閉曲線 B カラーデータ
閉曲線 C カラーデータ
閉曲線 D カラーデータ
閉曲線 E カラーデータ
閉曲線 F カラーデータ

変形データ (AD = 1)
変形データ (AD = 2)
変形データ (AD = 3)
⋮
変形データ (AD = n)

修正枠 (AD = 0)
修正枠 (AD = 1)
修正枠 (AD = 2)
⋮
修正枠 (AD = n)

【図 19】

閉曲線 A 作成データ	選択番号			11
閉曲線 B 作成データ	修正枠			12
閉曲線 C 作成データ	変形情報			13
閉曲線 D 作成データ				14
閉曲線 E 作成データ				15
閉曲線 F 作成データ				16
閉曲線 A カラーデータ				
閉曲線 B カラーデータ				
閉曲線 C カラーデータ				
閉曲線 D カラーデータ				
閉曲線 E カラーデータ				
閉曲線 F カラーデータ				
カラー状態フラグ (A)				
カラー状態フラグ (B)				
カラー状態フラグ (C)				
カラー状態フラグ (D)				
カラー状態フラグ (E)				
カラー状態フラグ (F)				
バックグラウンドカラー番号				

フロントページの続き

(51) Int. Cl. ⁶

識別記号

庁内整理番号

F I

技術表示箇所

9365-5L

G 0 6 F 15/72

4 0 0